

黑马程序员™  
[www.itheima.com](http://www.itheima.com)

传智播客旗下  
高端IT教育品牌

# Spring Cloud



# 目录 Contents

- ◆ Feign 声明式服务调用
- ◆ Hystrix 熔断器
- ◆ Gateway 网关

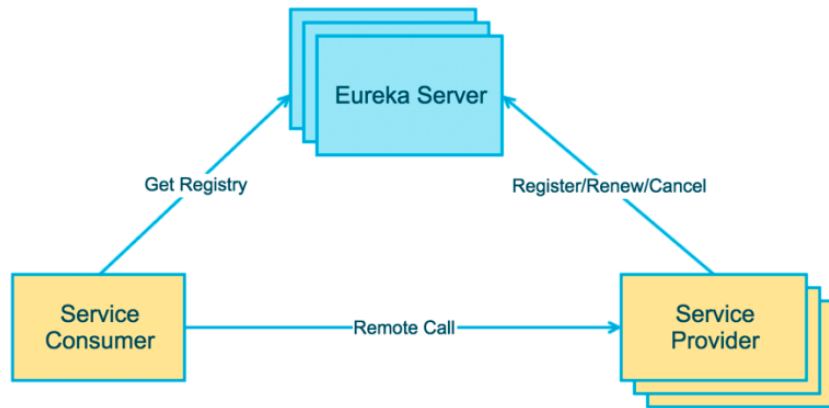
# Feign声明式服务调用

---

- Feign 概述
- Feign 快速入门
- Feign 其他功能

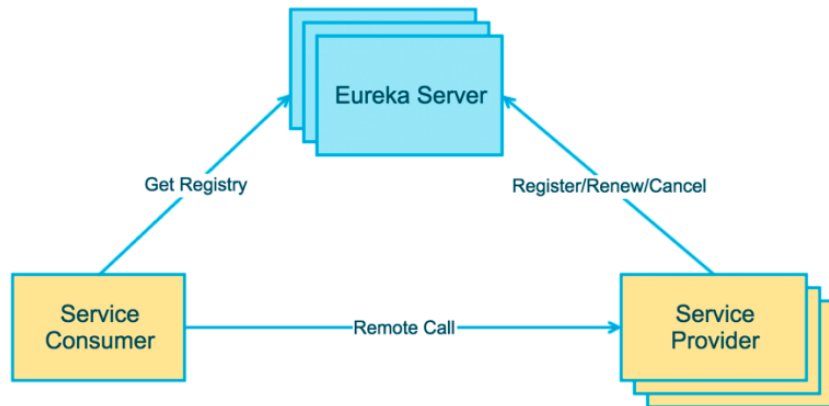
## Feign 概述

- Feign 是一个声明式的 REST 客户端，它用了基于接口的注解方式，很方便实现客户端配置。
- Feign 最初由 Netflix 公司提供，但不支持SpringMVC注解，后由 SpringCloud 对其封装，支持了SpringMVC注解，让使用者更易于接受。



## Feign 快速入门

1. 在消费端引入 open-feign 依赖
2. 编写Feign调用接口
3. 在启动类 添加 @EnableFeignClients 注解，开启Feign功能
4. 测试调用



# Feign声明式服务调用

---

- Feign 概述
- Feign 快速入门
- Feign 其他功能

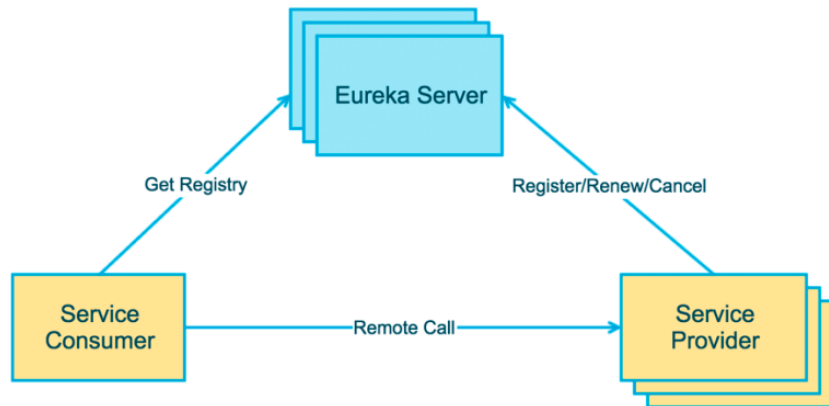
## Feign 其他功能 – 超时设置

- Feign 底层依赖于 Ribbon 实现负载均衡和远程调用。
- Ribbon默认1秒超时。
- 超时配置：

**ribbon:**

**ConnectTimeout:** 1000    #连接超时时间, 毫秒

**ReadTimeout:** 1000    #逻辑处理超时时间, 毫秒



## Feign 其他功能 – 日志记录

- Feign 只能记录 debug 级别的日志信息。

**logging:**

**level:**

**com.itheima:** debug

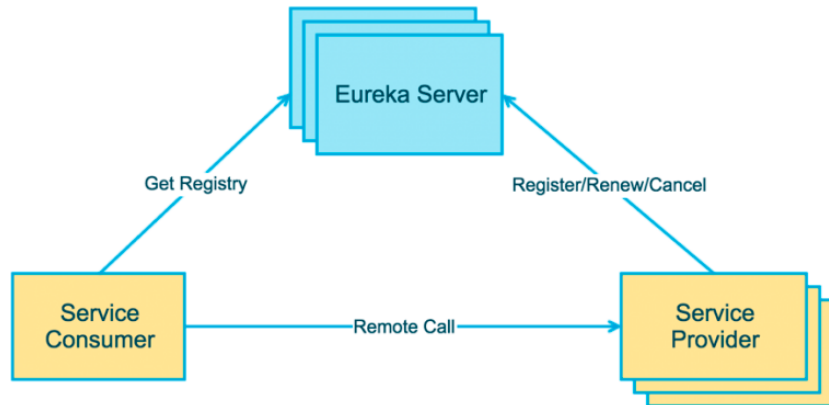
- 定义Feign日志级别Bean

**@Bean**

```
Logger.Level feignLoggerLevel() {  
    return Logger.Level.FULL;  
}
```

- 启用该Bean:

**@FeignClient**(configuration = XxxConfig.**class**)







# 目录 Contents

- ◆ Feign 声明式服务调用
- ◆ Hystrix 熔断器
- ◆ Gateway 网关

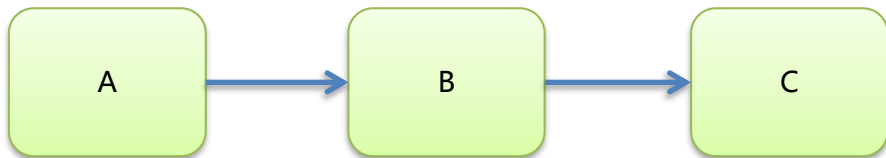
# Hystrix 熔断器

---

- Hystrix 概述
- Hystrix 降级
- Hystrix 熔断
- Hystrix 熔断监控

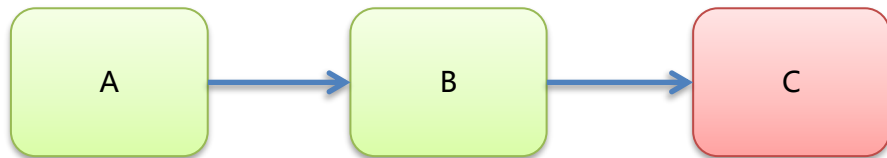
## Hystrix 概述

- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。



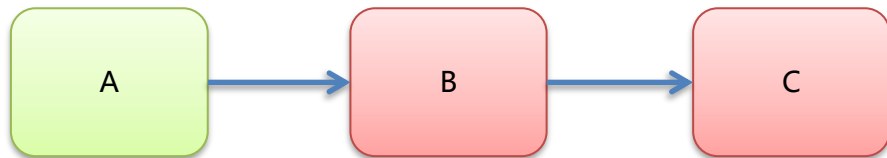
## Hystrix 概述

- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。



## Hystrix 概述

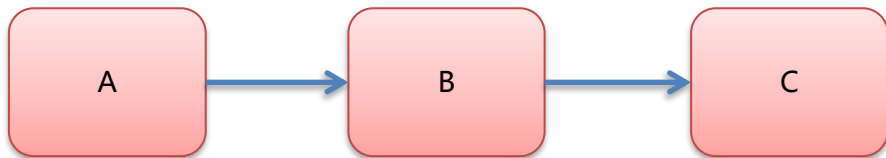
- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。





## Hystrix 概述

- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。





## Hystrix 概述

- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能
  - 隔离
  - 降级
  - 熔断
  - 限流



## Hystrix 概述

- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能
  - 隔离：
  - 降级
  - 熔断
  - 限流





## Hystrix 概述

- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能
  - 隔离：
    1. 线程池隔离
    2. 信号量隔离
  - 降级
  - 熔断
  - 限流



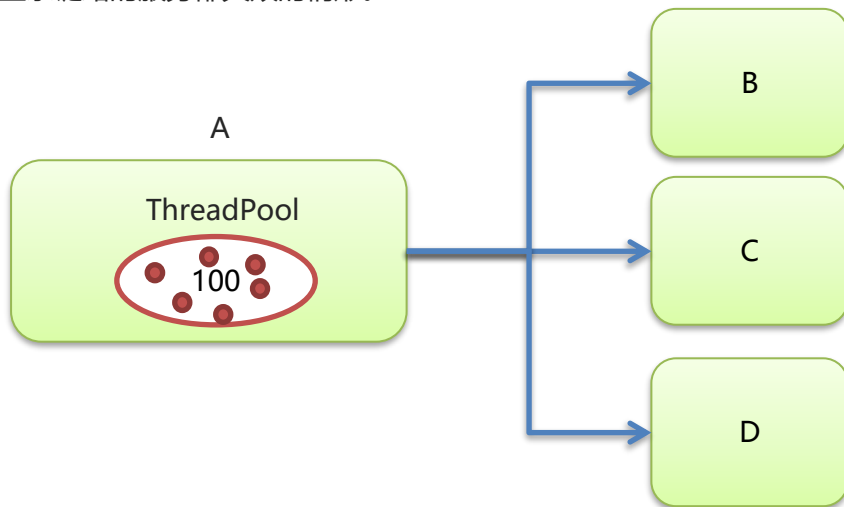
## Hystrix 概述

- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能
  - 隔离：
    1. 线程池隔离
    2. 信号量隔离
  - 降级
  - 熔断
  - 限流

## Hystrix 概述

- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能

- 隔离：
  1. 线程池隔离
  2. 信号量隔离
- 降级
- 熔断
- 限流





## Hystrix 概述

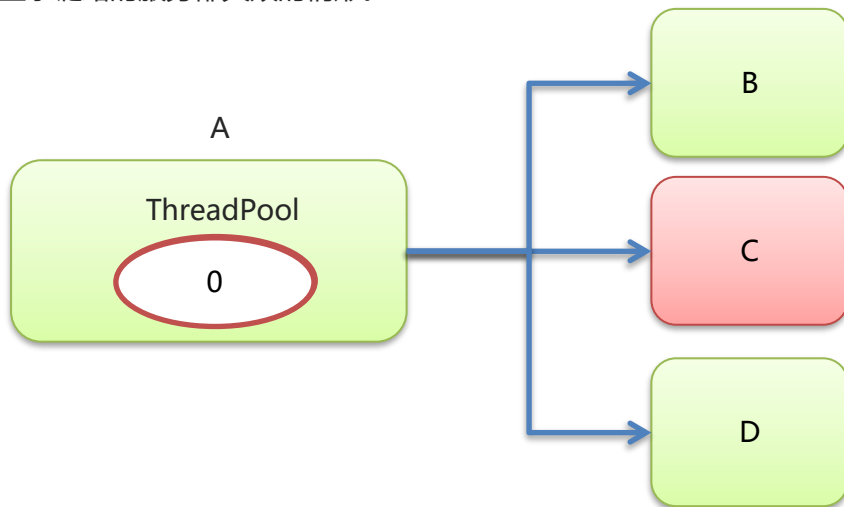
- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能

- 隔离

- 1. 线程池隔离

- 2. 信号量隔离

- 降级
  - 熔断
  - 限流



## Hystrix 概述

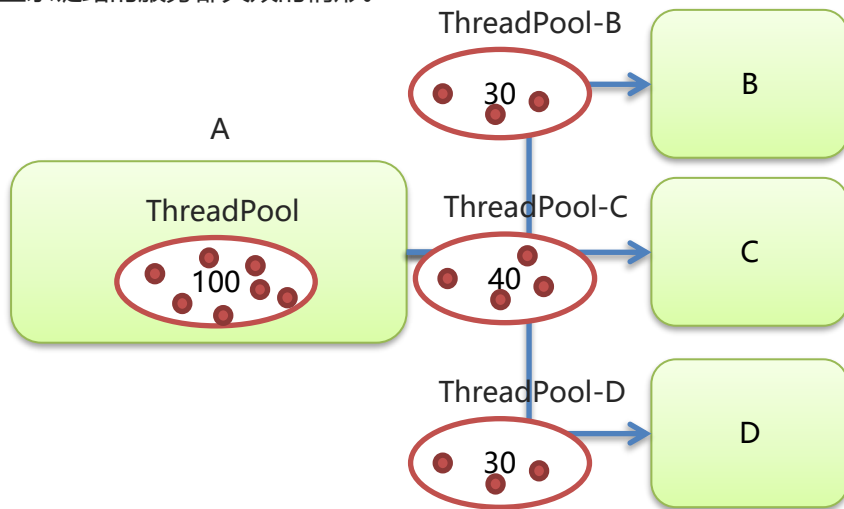
- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能

- 隔离

- 1. 线程池隔离

- 2. 信号量隔离

- 降级
  - 熔断
  - 限流



## Hystrix 概述

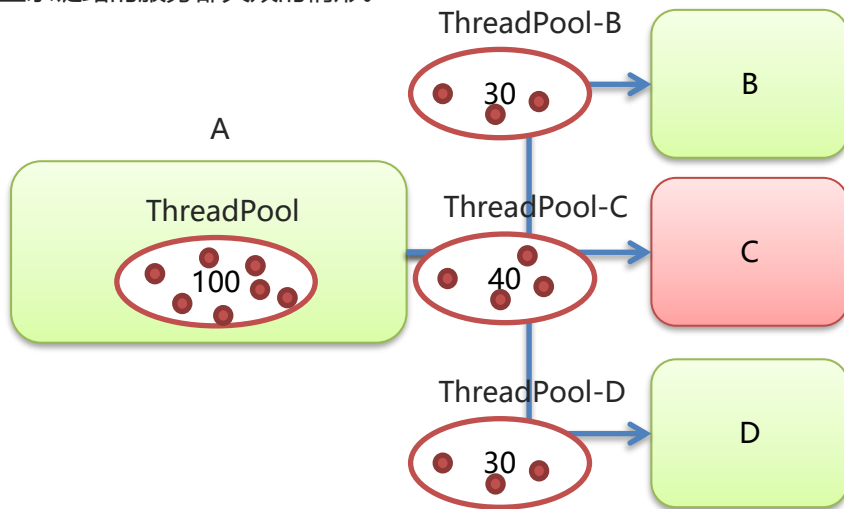
- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能

- 隔离

- 1. 线程池隔离

- 2. 信号量隔离

- 降级
  - 熔断
  - 限流



## Hystrix 概述

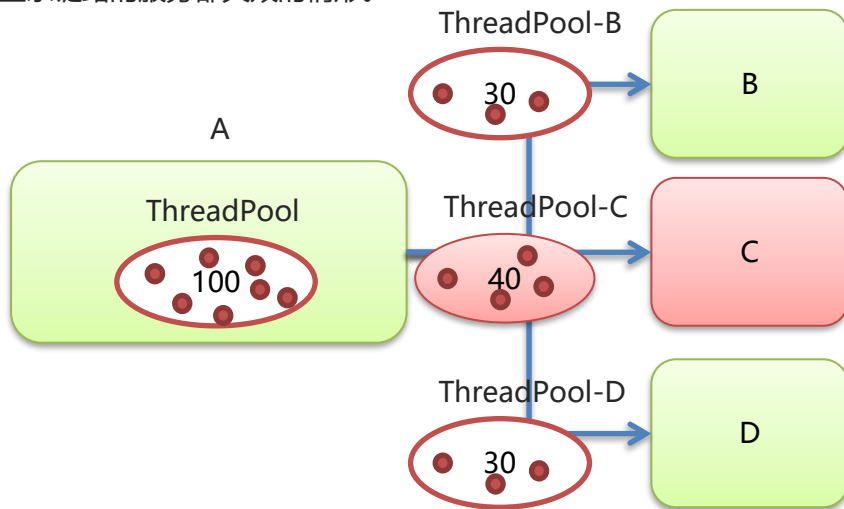
- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能

- 隔离

- 1. 线程池隔离

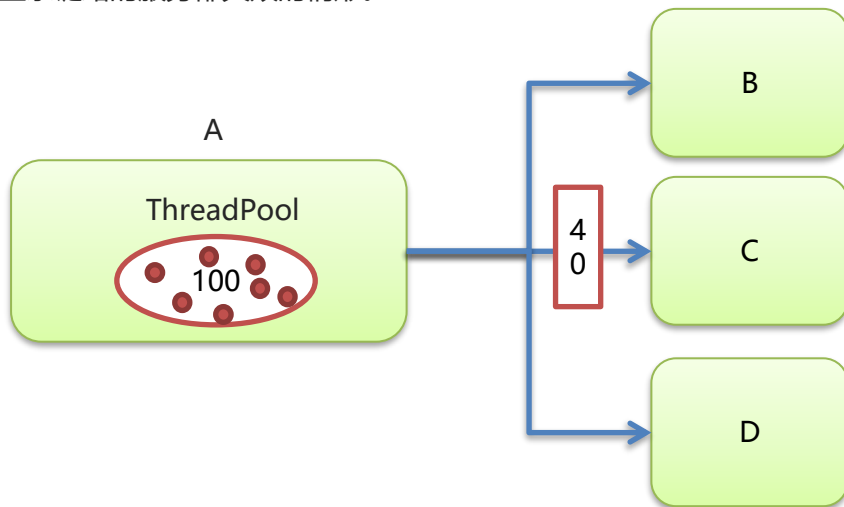
- 2. 信号量隔离

- 降级
  - 熔断
  - 限流



## Hystrix 概述

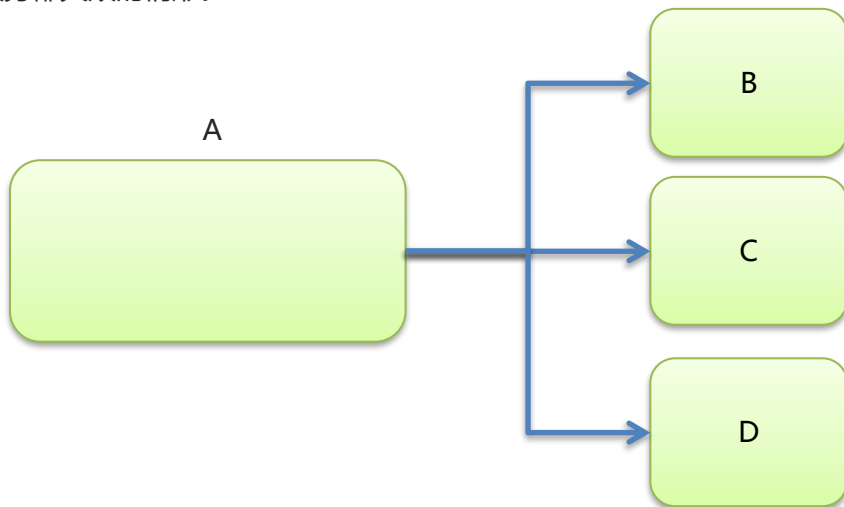
- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能
  - 隔离
    1. 线程池隔离
    2. 信号量隔离
  - 降级
  - 熔断
  - 限流





## Hystrix 概述

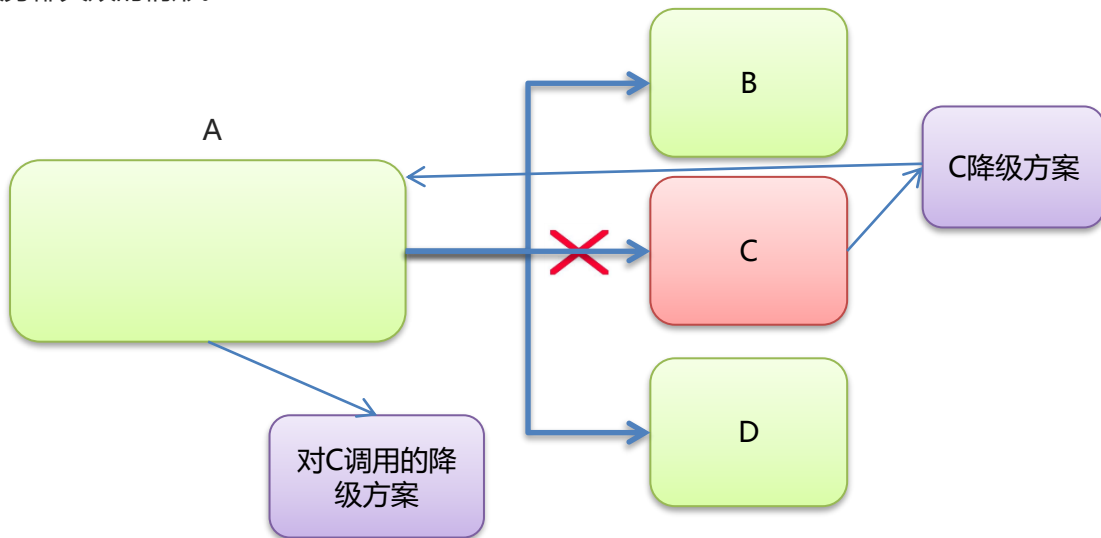
- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能
  - 隔离
    1. 线程池隔离
    2. 信号量隔离
  - 降级
  - 熔断
  - 限流





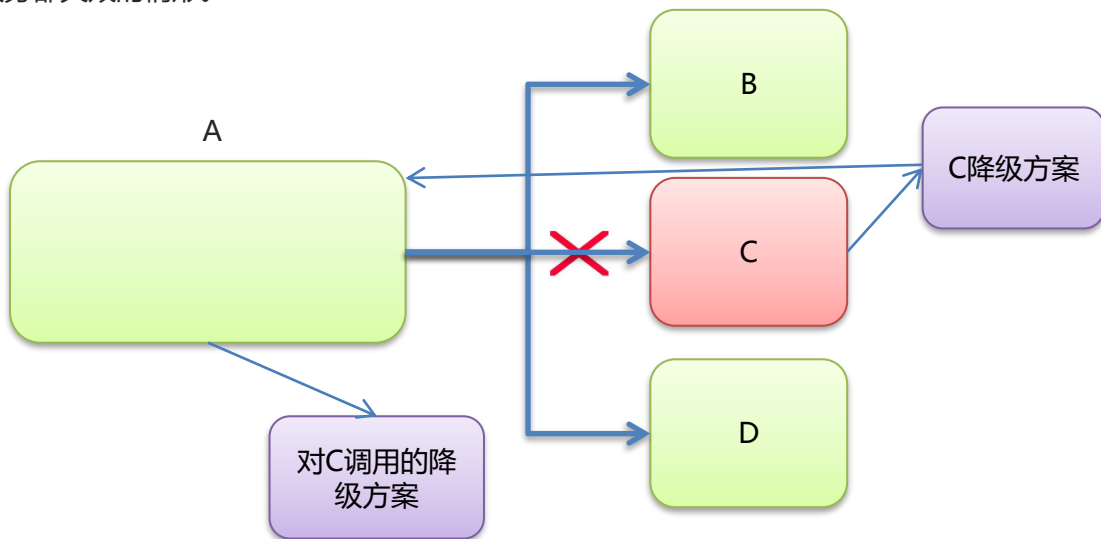
## Hystrix 概述

- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能
  - 隔离
    1. 线程池隔离
    2. 信号量隔离
  - **降级**：异常，超时
  - 熔断
  - 限流



## Hystrix 概述

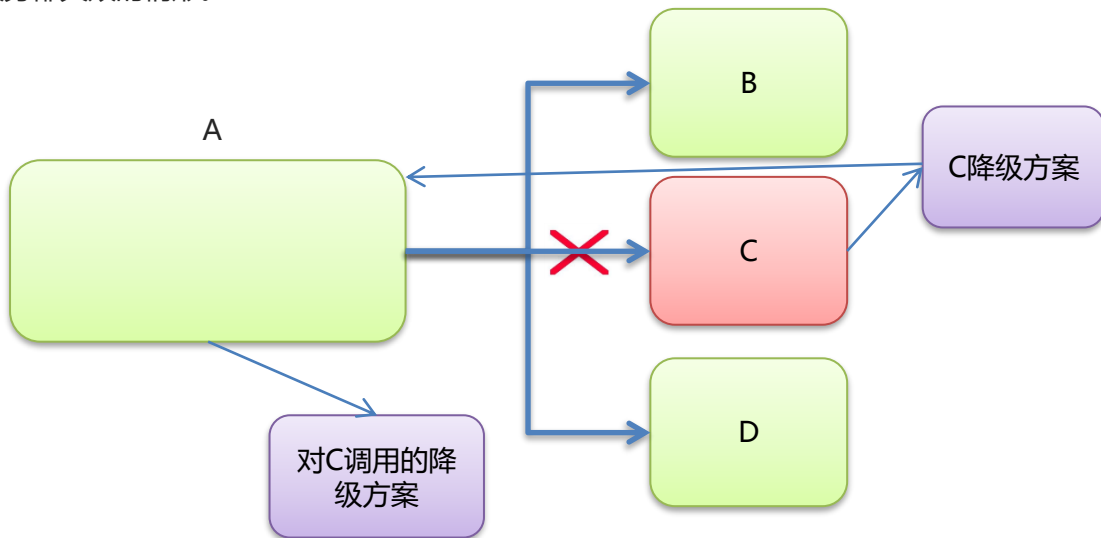
- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能
  - 隔离
    1. 线程池隔离
    2. 信号量隔离
  - 降级：异常，超时
  - **熔断**
  - 限流





## Hystrix 概述

- Hystrix 是 Netflix 开源的一个延迟和容错库，用于隔离访问远程服务、第三方库，防止出现级联失败（雪崩）。
- 雪崩：一个服务失败，导致整条链路的服务都失败的情形。
- Hystrix 主要功能
  - 隔离
    1. 线程池隔离
    2. 信号量隔离
  - 降级：异常，超时
  - 熔断
  - 限流



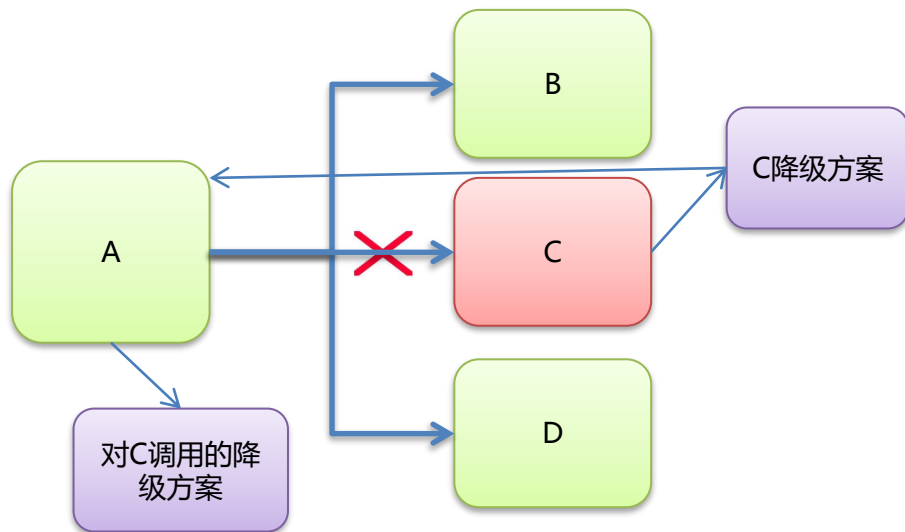
# Hystrix 熔断器

---

- Hystrix 概述
- Hystrix 降级
- Hystrix 熔断
- Hystrix 熔断监控

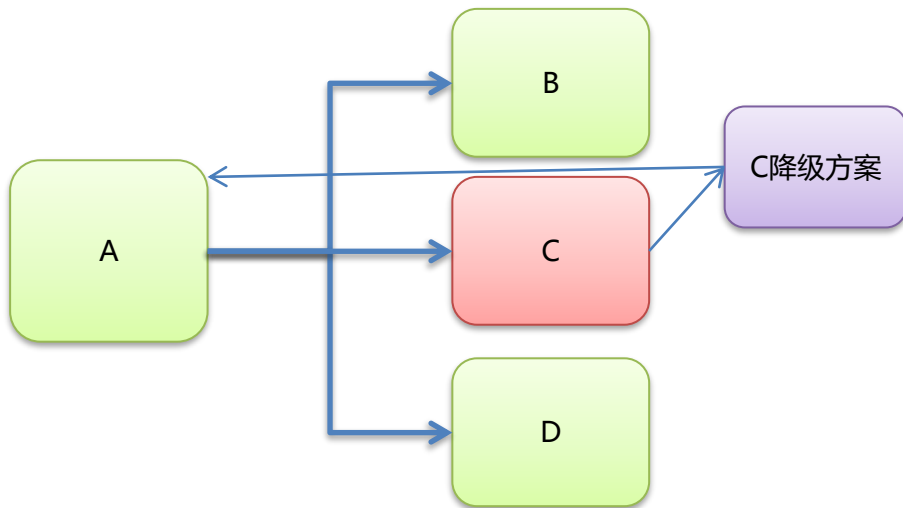
## Hystrix 降级

- Hystrix 降级：当服务发生异常或调用超时，返回默认数据



## Hystrix 降级 – 服务提供方

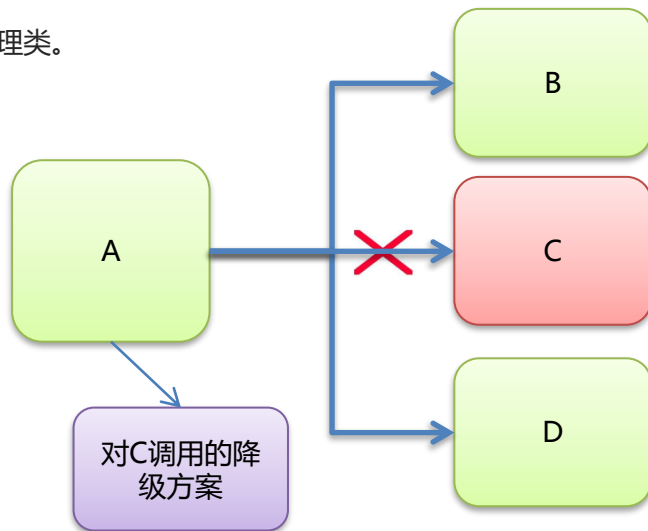
1. 在服务提供方，引入 hystrix 依赖
2. 定义降级方法
3. 使用 @HystrixCommand 注解配置降级方法
4. 在启动类上开启Hystrix功能: @EnableCircuitBreaker





## Hystrix 降级 – 服务消费方

1. feign 组件已经集成了 hystrix 组件。
2. 定义feign 调用接口实现类，复写方法，即 降级方法
3. 在 @FeignClient 注解中使用 fallback 属性设置降级处理类。
4. 配置开启 `feign.hystrix.enabled = true`





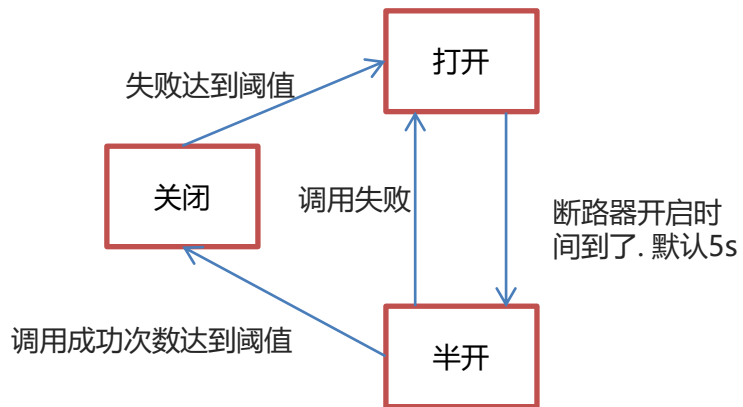
# Hystrix 熔断器

---

- Hystrix 概述
- Hystrix 降级
- Hystrix 熔断
- Hystrix 熔断监控

## Hystrix 熔断

- Hystrix 熔断机制，用于监控微服务调用情况，当失败的情况达到预定的阈值（5秒失败20次），会打开断路器，拒绝所有请求，直到服务恢复正常为止。



- `circuitBreaker.sleepWindowInMilliseconds`: 监控时间
- `circuitBreaker.requestVolumeThreshold`: 失败次数
- `circuitBreaker.errorThresholdPercentage`: 失败率



# Hystrix 熔断器

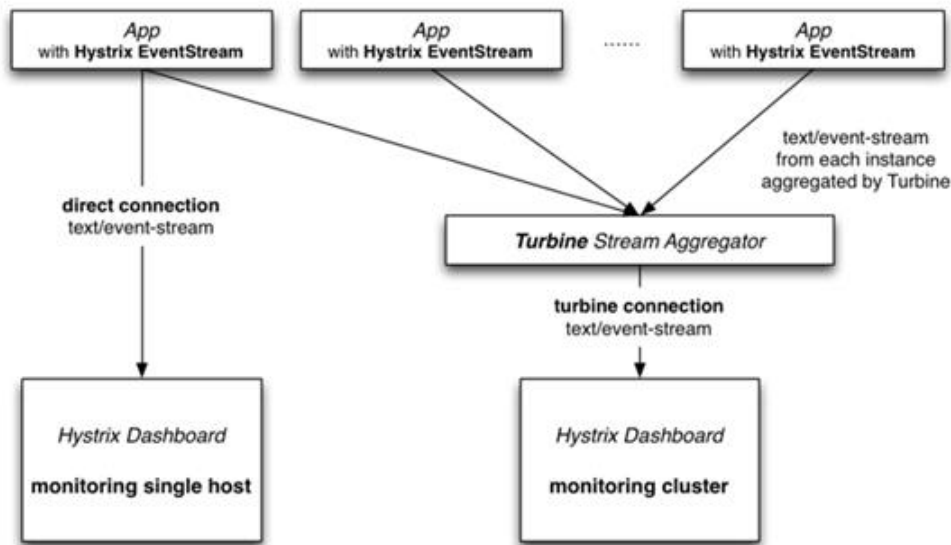
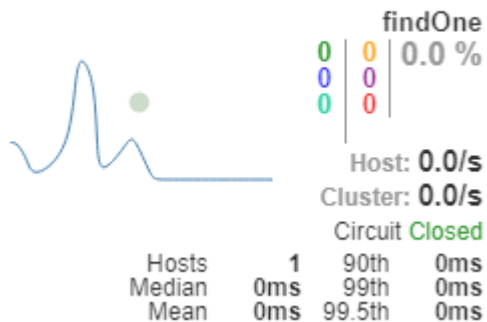
---

- Hystrix 概述
- Hystrix 降级
- Hystrix 熔断
- Hystrix 熔断监控

## Hystrix 熔断监控

- Hystrix 提供了 Hystrix-dashboard 功能，用于实时监控微服务运行状态。
- 但是Hystrix-dashboard只能监控一个微服务。
- Netflix 还提供了 Turbine，进行聚合监控。

**Circuit** Sort: [Error then Volume](#) | [Alpha](#)





# 目录 Contents

- ◆ Feign 声明式服务调用
- ◆ Hystrix 熔断器
- ◆ Gateway 网关

# Gateway 网关

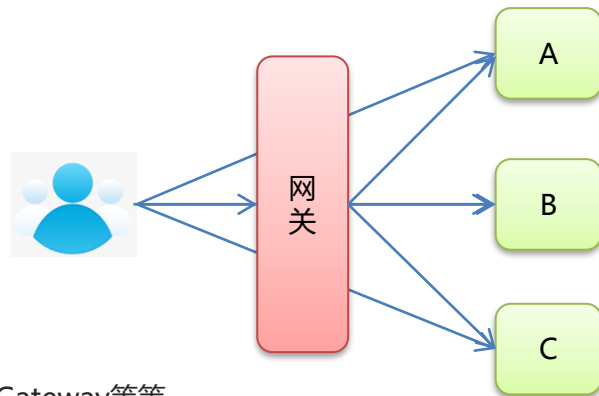
---

- 网关概述
- Gateway 网关快速入门
- Gateway 网关路由配置
- Gateway 网关过滤器



## 网关概述

- 网关旨在为微服务架构提供一种简单而有效的统一的API路由管理方式。
- 在微服务架构中，不同的微服务可以有不同的网络地址，各个微服务之间通过互相调用完成用户请求，客户端可能通过调用N个微服务的接口完成一个用户请求。
  - 存在的问题：
    - 客户端多次请求不同的微服务，增加客户端的复杂性
    - 认证复杂，每个服务都要进行认证
    - http请求不同服务次数增加，性能不高
- 网关就是系统的入口，封装了应用程序的内部结构，为客户端提供统一服务，一些与业务本身功能无关的公共逻辑可以在这里实现，诸如认证、鉴权、监控、缓存、负载均衡、流量管控、路由转发等
- 在目前的网关解决方案里，有Nginx+ Lua、Netflix Zuul 、Spring Cloud Gateway等等



# Gateway 网关

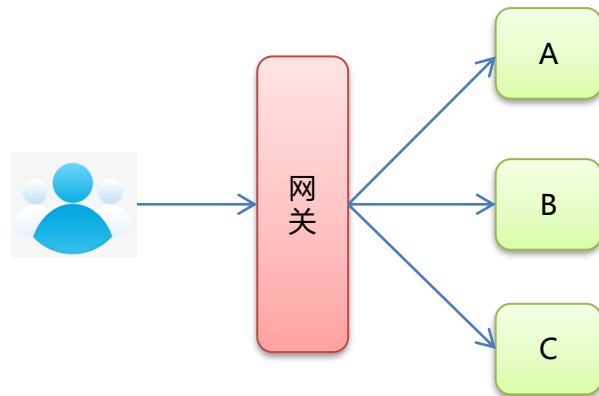
---

- 网关概述
- Gateway 网关快速入门
- Gateway 网关路由配置
- Gateway 网关过滤器



## Gateway 网关快速入门

1. 搭建网关模块
2. 引入依赖: starter-gateway
3. 编写启动类
4. 编写配置文件
5. 启动测试

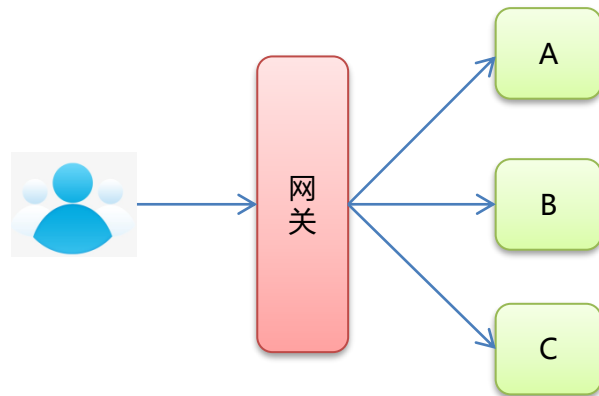


# Gateway 网关

---

- 网关概述
- Gateway 网关快速入门
- Gateway 网关路由配置
- Gateway 网关过滤器

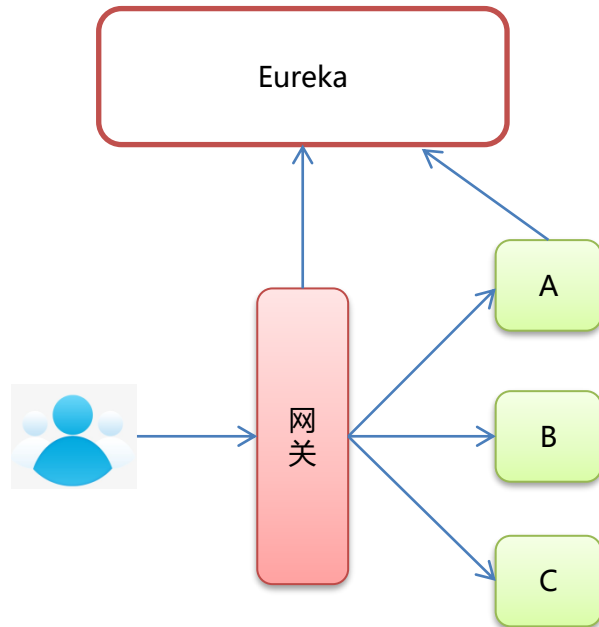
## Gateway 网关路由配置 – 静态路由





## Gateway 网关路由配置 – 动态路由

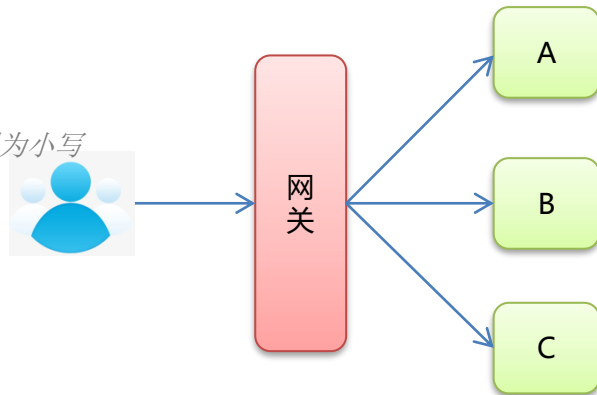
- 引入eureka-client配置
- 修改uri属性: uri: lb://服务名称





## Gateway 网关路由配置 – 微服务名称配置

```
spring:
  cloud:
    discovery:
      locator:
        enabled: true # 开启微服发现功能
        lower-case-service-id: true # 讲请求路径上的服务名配置为小写
```



# Gateway 网关

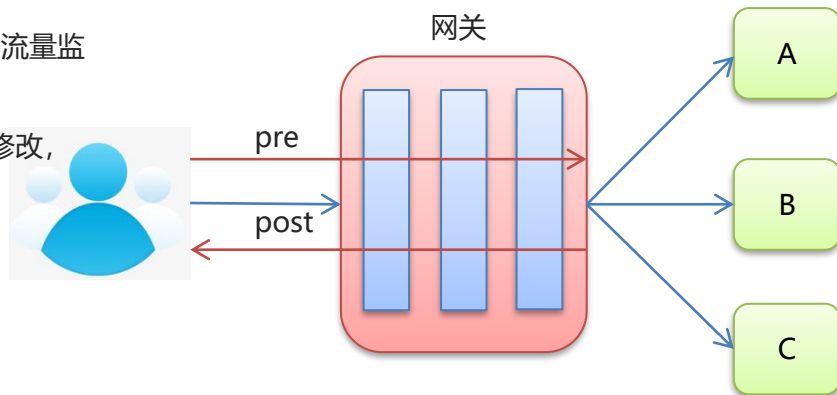
---

- 网关概述
- Gateway 网关快速入门
- Gateway 网关路由配置
- Gateway 网关过滤器



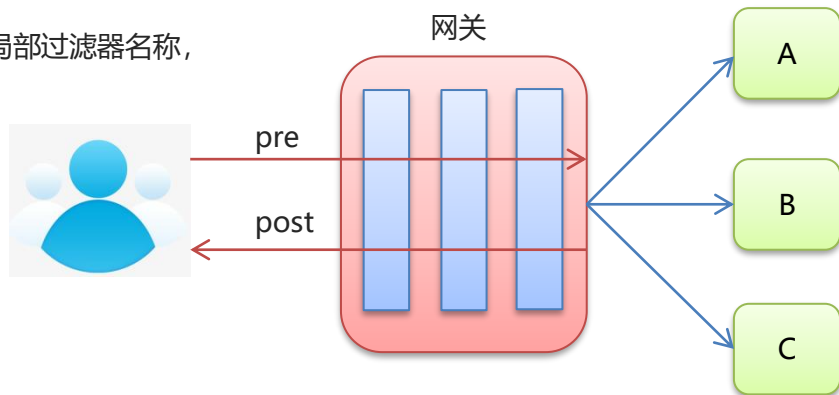
## Gateway 过滤器

- Gateway 支持过滤器功能，对请求或响应进行拦截，完成一些通用操作。
- Gateway 提供两种过滤器方式：“pre”和“post”
- pre 过滤器，在转发之前执行，可以做参数校验、权限校验、流量监控、日志输出、协议转换等。
- post 过滤器，在响应之前执行，可以做响应内容、响应头的修改，日志的输出，流量监控等。
- Gateway 还提供了两种类型过滤器
  - GatewayFilter：局部过滤器，针对单个路由
  - GlobalFilter：全局过滤器，针对所有路由



## Gateway 过滤器 – 局部过滤器

- GatewayFilter 局部过滤器，是针对单个路由的过滤器。
- 在Spring Cloud Gateway 组件中提供了大量内置的局部过滤器，对请求和响应做过滤操作。
- 遵循约定大于配置的思想，只需要在配置文件配置局部过滤器名称，并为其指定对应的值，就可以让其生效。







## Gateway 过滤器 – 全局过滤器

- GlobalFilter 全局过滤器，不需要在配置文件中配置，系统初始化时加载，并作用在每个路由上。
- Spring Cloud Gateway 核心的功能也是通过内置的全局过滤器来完成。
- 自定义全局过滤器步骤：
  1. 定义类实现 GlobalFilter 和 Ordered接口
  2. 复写方法
  3. 完成逻辑处理





传智播客旗下高端IT教育品牌