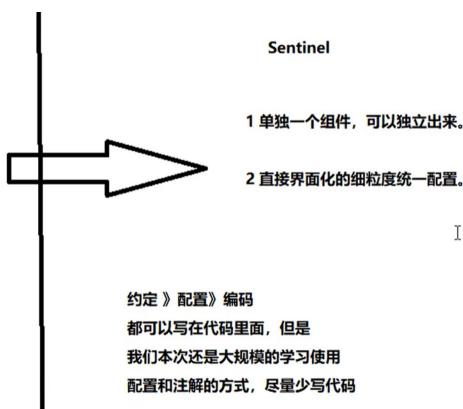


熔断和限流
hystrix的阿里版

Hystrix

- 1 需要我们程序员自己手工搭建监控平台
- 2 没有一套web界面可以给我们进行更加细粒度化得配置
流控、速率控制、服务熔断、服务降级。



Sentinel: 分布式系统的流量防卫兵

Sentinel 是什么？

随着微服务的流行，服务和服务之间的稳定性变得越来越重要。Sentinel 以流量为切入点，从流量控制、熔断降级、系统负载保护等多个维度保护服务的稳定性。

Sentinel 具有以下特征：

- **丰富的应用场景：** Sentinel 承接了阿里巴巴近 10 年的双十一大促流量的核心场景，例如秒杀（即突发流量控制在系统容量可以承受的范围）、消息削峰填谷、集群流量控制、实时熔断下游不可用应用等。
- **完备的实时监控：** Sentinel 同时提供实时的监控功能。您可以在控制台中看到接入应用的单台机器秒级数据，甚至 500 台以下规模的集群的汇总运行情况。
- **广泛的开源生态：** Sentinel 提供开箱即用的与其它开源框架/库的整合模块，例如与 Spring Cloud、Dubbo、gRPC 的整合。您只需要引入相应的依赖并进行简单的配置即可快速地接入 Sentinel。
- **完善的 SPI 扩展点：** Sentinel 提供简单易用、完善的 SPI 扩展接口。您可以通过实现扩展接口来快速地定制逻辑。例如定制规则管理、适配动态数据源等。

-----流控-----

快速失败

表示1秒钟内查询1次就是OK，若超过次数1，就直接-快速失败，报默认错误

新增流控规则

资源名: /testA

针对来源: default

阈值类型: QPS 线程数

单机阈值: 1

是否集群:

流控模式: 直接 关联 链路

流控效果: 快速失败 Warm Up 排队等待

关闭高级选项

QPS:每秒数量

线程数，浏览器处理超过阈值进行限流

关联B坏了,A不可以

编辑流控规则

资源名 /testA

针对来源 default

阈值类型 QPS 线程数 单机阈值 1

是否集群

流控模式 直接 关联 链路

关联资源 /testB

流控效果 快速失败 Warm Up 排队等待

[关闭高级选项](#)

预热，假如从0到10w访问数一下子接受不了，那么就需要预热

Warm Up (`RuleConstant.CONTROL_BEHAVIOR_WARM_UP`) 方式，即预热/冷启动方式。当系统长期处于低水位的情况下，当流量突然增加时，直接把系统拉升到高水位可能瞬间把系统压垮。通过“冷启动”，让通过的流量缓慢增加，在一定时间内逐渐增加到阈值上限，给冷系统一个预热的时间，避免冷系统被压垮。详细文档可以参考[流量控制 - Warm Up 文档](#)，具体的例子可以参见

[WarmUpFlowDemo](#)。

冷加载因子为3

下图表述阈值从阈值/3开始，也就是刚开始阈值为3逐渐（5秒后）变为10

刚开始访问的时候有限流的情况，后续会逐渐放大，到阈值10，就不会出现限流信息提示了(1秒点不了10下)

资源名 /testA

针对来源 default

阈值类型 QPS 线程数 单机阈值 10

是否集群

流控模式 直接 关联 链路

关联资源 /testB

流控效果 快速失败 Warm Up 排队等待

预热时长 5

[关闭高级选项](#)

排队等待，进来需要排队

下图为1秒1次访问，在1秒内访问会进行排队等待，如果等待时间超过500ms，那么则返回报错信息

资源名 /testA

针对来源 default

阈值类型 QPS 线程数 单机阈值 1

是否集群

流控模式 直接 关联 链路

关联资源 /testB

流控效果 快速失败 Warm Up 排队等待

超时时间 500

[关闭高级选项](#)

降级

RT (平均响应时间, 秒级)

平均响应时间 超出阈值 且 在时间窗口内通过的请求 ≥ 5 , 两个条件同时满足后触发降级
窗口期过后关闭断路器

RT最大4900 (更大的需要通过-Dcsp.sentinel.statistic.max.rt=XXXX才能生效)

上图表示在200毫秒以内处理完成, 如果搞不定, 在未来的一秒之内进行熔断

按照上述配置,

永远一秒钟打进来10个线程 (大于5个了) 调用testD, 我们希望200毫秒处理完本次任务,
如果超过200毫秒还没处理完, 在未来1秒钟的时间窗口内, 断路器打开(保险丝跳闸)微服务不可用, 保险丝跳闸断电了

他这个方法是睡一秒。假如规定2秒内完成, 不管几个线程数 (目前10个) 也不会影响运行

```
@GetMapping("/testD")
public String testD() {
    try {
        TimeUnit.SECONDS.sleep(timeout);
    } catch (InterruptedException e) {
        ...
    }
}
```

假如限制为200毫秒, 而它响应的时间为1秒, 没有在规定时间内完成, 那么不管几个线程
(测试4个) 也会进行降级。

经过测试可以得出结论, 不论进程数为多少, 看你的响应时间如果 \leq 你设置的规定时间那么,
服务不会进行熔断, 反之进行熔断, 就算我设置规定时间为1秒, 10个线程也是可以访问的,
而100个就不行了, 因为实际响应时间已经超过了1秒钟大于我设置的时间就会发生服务熔断

的现象

异常比例

sentinel没有半开状态，上图中的事件窗口就表示降级所持续的时间，即使后面不管什么请求进入在规定时间窗口后，降级就会关闭，路径正常访问

在1秒内访问错误率达到百分之50就是熔断，一秒一线程也会熔断，这也是一个问题



异常数



时间窗口不能低于60秒

-----热点规则-----



@SentinelResource(value = "testHotKey",blockHandler = "deal_testHotKey")

这个注解相当于兜底的方法，相当于hystrix的@HystrixCommand注解

```
@GetMapping("/testHotKey")
@SentinelResource(value = "testHotKey", blockHandler = "deal_testHotKey")
public String testHotKey(@RequestParam(value = "p1", required = false)String p1,
    @RequestParam(value = "p2", required = false)String p2)
    return "----testHotKey";
```

对应索引下标0->p1,1->p2
对应资源名称

索引为0是只要有p1都会管，没p1都不会管

编辑热点规则

资源名	testHotKey
限流模式	QPS 模式
参数索引	0
单机阈值	1
统计窗口时长	1 秒
是否集群	<input type="checkbox"/>

参数例外项

参数类型	参数值	限流阈值	操作
参数值	5	java.lang.String	200

[+添加](#)

[关闭高级选项](#)

高级选项，当我p1=5的时候会进入特殊模式，限流阈值不在是1000而是200

热点只负责配置类的错，不负责Java运行时候产生的错误，比如除数为0

fallback管运行异常

blockHandler管配置违规

若 blockHandler 和 fallback 都进行了配置，则被限流降级而抛出 BlockException 时只会进入 blockHandler 处理逻辑。