



石臻

1 专家

码龄8年

Java领域优质创作者

114

1917

9418

51万+

原创

周排名

总排名

访问

等级

6148

1万+

324

204

708

积分

粉丝

获赞

评论

收藏

私信

关注

搜博主文章

热门文章

请将文件MP_verify_N8no8dpNO9n7wuBZ.txt上传至XXX指向的web服务器（或虚拟主机）的目录

35465

基于Redis实现DelayQueue延迟队列设计方案

26846

【Seata】分布式事务框架Seata踩坑集锦

21216

Spring源码解析(十一)Spring扩展接口InstantiationAwareBeanPostProcessor解析

19459

一场HttpClient调用未关闭流引发的问题

16105

分类专栏

Nacos配置管理

付费

14篇

kafka-manager

3篇

Althas

3篇

笔记

Logi-KafkaManager

jvm

2篇

最新评论

Kafka的灵魂伴侣Logi-KafkaManger三之...

石臻: 好的

Kafka的灵魂伴侣Logi-KafkaManger三之...

Lansonli: 还可以这样? 秀! 支持博主, 大佬有兴趣也可以看下我的博客, 说不定有...

Kafka的灵魂伴侣Logi-KafkaManger三之...

石臻: 😊

Kafka的灵魂伴侣Logi-KafkaManger三之...

石臻: 😊😊😊😊😊

Kafka的灵魂伴侣Logi-KafkaManger三之...

不正经的kimoi君: 学习了, 谢谢分享~

最新文章

Kafka的灵魂伴侣Logi-KafkaManger二之kafka针对Topic粒度的配额管理(限流)

Kafka的灵魂伴侣Logi-KafkaManger三之运维管控-集群列表

Kafka的灵魂伴侣Logi-KafkaManger一之集群的接入及相关概念讲解

2021年 9篇

2020年 22篇

2019年 38篇

2018年 32篇

2017年 5篇

2016年 3篇

2015年 3篇

2014年 3篇

目录

说明

@KafkaListener详解

id 监听器的id

groupId 消费组名

topics 指定要监听哪些topic(与topicPattern 匹配Topic进行监听(与topicPartitions 显式分区分配

errorHandler 异常处理

containerFactory 监听器工厂

clientIdPrefix 客户端前缀

concurrency并发数

properties 配置其他属性

@KafkaListener使用

【spring-kafka】@KafkaListener详解与使用

原创

石臻

2021-01-07 17:09:27

2982

收藏 9

版权

分类专栏:

spring-kafka

Nacos配置管理

本专栏深入浅出的解析了Nacos配置管理模块的原理 跟着源码带你了解Nacos这一强大的开源项目 并且画有多张流程图提供了解, 非常清晰 ...

石臻

¥9.90

订阅专栏

说明

- 从2.2.4版开始，您可以直接在注释上指定Kafka使用者属性，这些属性将覆盖在使用者工厂中配置的具有相同名称的所有属性。您不能通过这种方式指定group.id和client.id属性。他们将被忽略；
- 可以使用#{...}或属性占位符 (\${...}) 在SpEL上配置注释上的大多数属性。比如：

```
1 @KafkaListener(id = "consumer-id",topics = "SHI_TOPIC1",concurrency = "${listen.
2 clientIdPrefix = "myClientId")
```

属性 concurrency 将会从容器中获取 listen.concurrency 的值.如果不存在就默认用3

@KafkaListener详解

id 监听器的id

①. 消费者线程命名规则

填写：

```
2020-11-19 14:24:15 c.d.b.k.KafkaListeners 120 [INFO] 线程:Thread[ consumer-id5-1-C-1 ,5,main]-
groupId:BASE-DEMO consumer-id5 消费
```

没有填写ID：

```
2020-11-19 10:41:26 c.d.b.k.KafkaListeners 137 [INFO] 线
程:Thread[ org.springframework.kafka.KafkaListenerEndpointContainer#0-0-C-1 ,5,main]
consumer-id7
```

②.在相同容器中的监听器ID不能重复

否则会报错

```
1 Caused by: java.lang.IllegalStateException: Another endpoint is already registered
```

③.会覆盖消费者工厂的消费组GroupId

假如配置文件属性配置了消费组 kafka.consumer.group-id=BASE-DEMO
正常情况它是该容器中的默认消费组
但是如果设置了 @KafkaListener(id = "consumer-id7", topics = {"SHI_TOPIC3"})
那么当前消费者的消费组就是 consumer-id7 ；

当然如果你不想要他作为groupId的话 可以设置属性 idIsGroup = false ;那么还是会使用默认的GroupId;

④. 如果配置了属性groupId,则其优先级最高

```
1 @KafkaListener(id = "consumer-id5",idIsGroup = false,topics = "SHI_TOPIC3",groupId
```

例如上面代码中最终这个消费者的消费组 groupId 是 "groupId-test"

该id属性（如果存在）将用作Kafka消费者group.id属性,并覆盖消费者工厂中的已配置属性（如果存在）您还可以以groupId显式设置或将其设置idIsGroup为false，以恢复使用使用者工厂的先前行为group.id。

groupId 消费组名

指定该消费组的消费组名; 关于消费组名的配置可以看看上面的 id 监听器的id

如何获取消费者 group.id

在监听器中调用 KafkaUtils.getConsumerGroupId() 可以获得当前的groupId; 可以在日志中打印出来; 可以知道是哪个客户端消费的;

topics 指定要监听哪些topic(与topicPattern、 topicPartitions 三选一)

可以同时监听多个

```
topics = {"SHI_TOPIC3","SHI_TOPIC4"}
```

topicPattern 匹配Topic进行监听(与topics、 topicPartitions 三选一)

topicPartitions 显式分区分配

可以为监听器配置明确的主题和分区（以及可选的初始偏移量）

```
1 @KafkaListener(id = "thing2", topicPartitions =
2 { @TopicPartition(topic = "topic1", partitions = { "0", "1" })),
3 @TopicPartition(topic = "topic2", partitions = "0",
4 partitionOffsets = @PartitionOffset(partition = "1", initialOffset = "
5 })
6 public void listen(ConsumerRecord<?, ?> record) {
7 ...
8 }
```

上面例子意思是 监听 topic1 的0,1分区; 监听 topic2 的第0分区,并且第1分区从offset为100的开始消费;

errorHandler 异常处理

实现 KafkaListenerErrorHandler ; 然后做一些异常处理:

```
1 @Component
2 public class KafkaDefaultListenerErrorHandler implements KafkaListenerErrorHandler
3 {
4     @Override
5     public Object handleError(Message<?> message, ListenerExecutionFailedException
6         return null;
7 }
8
9 @Override
10 public Object handleError(Message<?> message, ListenerExecutionFailedException
11 //do someting
12     return null;
13 }
```

调用的时候 填写beanName;例如 errorHandler="KafkaDefaultListenerErrorHandler"

containerFactory 监听器工厂

指定生成监听器的工厂类;

例如我写一个 批量消费的工厂类

```
1  /**
2   * 监听器工厂 批量消费
3   * @return
4   */
5  @Bean
6  public KafkaListenerContainerFactory<ConcurrentMessageListenerContainer<Integer
7      ConcurrentKafkaListenerContainerFactory<Integer, String> factory =
8      new ConcurrentKafkaListenerContainerFactory<>();
9      factory.setConsumerFactory(kafkaConsumerFactory());
10     //设置为批量消费，每个批次数量在Kafka配置参数中设置ConsumerConfig.MAX_POLL_RECORDS_
11     factory.setBatchListener(true);
12     return factory;
13 }
```

使用 containerFactory = "batchFactory"

clientIdPrefix 客户端前缀

会覆盖消费者工厂的 kafka.consumer.client-id 属性; 最为前缀后面接 -n n是数字

concurrency并发数

会覆盖消费者工厂中的concurrency ,这里的并发数就是多线程消费; 比如说单机情况下,你设置了3; 相当于就是启动了3个客户端来分配消费分区;分布式情况 总线程数=concurrency*机器数量; 并不是设置越多越好,具体如何设置请看 属性concurrency的作用及配置(RoundRobinAssignor 、RangeAssignor)

```
1  /**
2   * 监听器工厂
3   * @return
4   */
5  @Bean
6  public KafkaListenerContainerFactory<ConcurrentMessageListenerContainer<Integer
7      ConcurrentKafkaListenerContainerFactory<Integer, String> factory =
8      new ConcurrentKafkaListenerContainerFactory<>();
9      factory.setConsumerFactory(kafkaConsumerFactory());
10     factory.setConcurrency(6);
11     return factory;
12 }
```

```
1  @KafkaListener(id = "consumer-id5",idIsGroup = false,topics = "SHI_TOPIC3", con
```

虽然使用的工厂是 concurrencyFactory (concurrency配置了6); 但是他最终生成的监听器数量 是1;

properties 配置其他属性

kafka中的属性看 org.apache.kafka.clients.consumer.ConsumerConfig ; 同名的都可以修改掉;

用法

```
1  @KafkaListener(id = "consumer-id5",idIsGroup = false,topics = "SHI_TOPIC3", con
2      , clientIdPrefix = "myClientId5",groupId = "groupId-test",
3      properties = {
4          "enable.auto.commit:false","max.poll.interval.ms:6000" },errorH
```

@KafkaListener使用

KafkaListenerEndpointRegistry

```
1  @Autowired
2  private KafkaListenerEndpointRegistry registry;
3  //.... 获取所有注册的监听器
4  registry.getAllListenerContainers();
5
```

设置入参验证器

当您将Spring Boot与验证启动器一起使用时, 将LocalValidatorFactoryBean自动配置: 如下

```
1  @Configuration
2  @EnableKafka
3  public class Config implements KafkaListenerConfigurer {
4
5      @Autowired
6      private LocalValidatorFactoryBean validator;
7      ...
8
9      @Override
10     public void configureKafkaListeners(KafkaListenerEndpointRegistrar registrar) {
11         registrar.setValidator(this.validator);
12     }
13 }
```


使用


```
1  @KafkaListener(id="validated", topics = "annotated35", errorHandler = "validationEr
2      containerFactory = "kafkaJsonListenerContainerFactory")
3  public void validatedListener(@Payload @Valid ValidatedClass val) {
4      ...
5  }
6
7  @Bean
8  public KafkaListenerErrorHandler validationErrorHandler() {
9      return (m, e) -> {
10         ...
11     };
12 }
```

TODO...

Springboot2 (30) 集成kafka--详细讲解@KafkaListener cowbin2012的专栏 3万+
源码地址 springboot2教程系列 写性能非常高, 因此, 经常会碰到Kafka消息队列拥堵的情况 经测试, 如果该topic只有一个...

spring-kafka通过@KafkaListener实现消费者监听流程分析 自律使我自由 3983
首先通过@EnableKafka注解, 注入KafkaBootstrapConfiguration类 @Target(ElementType.TYPE) @Retention(RetentionP...

 优质评论可以帮助作者获得更高权重

 Java劝退师、： 大佬的文章一直是我坚持写作的动力。 最近我也在学习写博客,有空来看看我呀, 一起互相学习。 期待你的关注与支持 3 月前 回复

相关推荐

深入理解 spring-kafka 监听器创建与运行以及消息处理流程 4-5
好久没有写博客了,正好最近在工作的时候,使用 spring-kafka 消费消息时候遇到一个关于批量消息处理的问题,通过阅读 spri...

spring boot 集成kafka (多线程,消费者使用kafka的原生api实现,因为@KakfkaListe... qq_40633152的博客 7514
原文链接: http://www.mamicode.com/info-detail-2078498.html 1 #kafka 2 kafka.consumer.zookeeper.connect="2181 3 k...

(1条消息) 【Spring-kafka】@KafkaListener详解与使用_石臻臻-CSDN博客		
官方文档: https://docs.spring.io/spring-kafka/reference/html/ @KafkaListener The @KafkaListener annotation is used to de...		
Spring Kafka @KafkaListener和它的朋友们	来啊 快活啊	3342
Spring-Kafka (六) —— @KafkaListener的花式操作 Spring Boot 中使用@KafkaListener并发批量接收消息		
springboot+@KafkaListener 消费者参数详解	asd56296266的博客	1万+
1.1 consumer参数详解 BOOTSTRAP_SERVERS_CONFIG kafka ip+port REQUEST_TIMEOUT_MS_CONFIG 请求超时...		
@KafkaListener注解学习	WorldMvp的专栏	2963
本文只是介绍一下@KafkaListener的工作流程，并不详细深入。 一、@KafkaListener 注解使用方式 spring-kafka使用基于...		
SpringKafka动态指定@KafkaListener的topics和groupId	u014259503的博客	2461
1.@KafkaListener @KafkaListener是kafka的消费者，topics是其主题名，groupId是组名； 属性值一般只支持常量，再集群...		
springboot2.x +kafka使用和源码分析九(KafkaListenerEndpointRegistry暂停启动... F_Hello_World的博客		1199
我们在运行中如果需要暂停启动容器时可以通过此类KafkaListenerEndpointRegistry来处理。 KafkaListenerEndpointRegist...		
Kafka动态关闭、开启监听	小小码农	5383
在实际生产开发中经常会有这样的场景，因为某些场景需要暂时关闭kafka的监听，比如重刷缓存等，等刷好之后再度开启k...		
Spring Boot 中使用@KafkaListener并发批量接收消息	russle的专栏	10万+
kakfa是 我们在项目开发中经常使用的消息中间件。由于它的写性能非常高，因此，经常会碰到Kafka消息队列拥堵的情况...		
kafka API详解	zz的博客	2万+
1. 生产者apiProducer是Kafka三大组件中的一个，用于发送消息到kafka集群中Producer提供了丰富的配置（见后面的配置...		
springboot+@KafkaListener 消费者参数详解	m0_37364073的博客	3217
1.1 consumer参数详解 BOOTSTRAP_SERVERS_CONFIG kafka ip+port REQUEST_TIMEOUT_MS_CONFIG 请求超时...		
Spring Boot整合Kafka的简单用例(@KafkaListener注解)	yyhnap的博客	3万+
第一步、启动zookeeper server和kafka server 启动zookeeper server: bin/zookeeper-server-start.sh config/zookeeper.pro...		
Spring Boot 中使用@KafkaListener批量接收消息ack	weixin_34347651的博客	2978
2019独角兽企业重金招聘Python工程师标准>>> ...		
springboot kafka group.id多消费组配置	关注微信公众号“虾米聊吧” 获取所有资料干货	2万+
很早之前就使用了springboot + kafka组合配置，但是之前使用的spring-kafka (1.1.7) 版本较低，所以只能通过 spring.kafk...		
spring-kafka的线程模型与spring.kafka.listener.concurrency参数	王鸿飞的专栏	1万+
在spring应用中如果需要订阅kafka消息，通常情况下我们不会直接使用kafka-client, 而是使用更方便的一层封装spring-kafk...		
kafka springboot+@KafkaListener 自动提交的简单使用	asd56296266的博客	5534
spring-boot 版本 1.5.12 依赖使用spring-kafka1.3.3 （对应kafka-clients版本0.11.0.0，请使用于kafka版本对应版本的依赖） ...		
spring集成kafka，实现一个topic可以被多个group消费	人生谏只如初见，何事秋风悲画扇	2万+
由于新公司是做物联网的，公司刚起步，没什么项目，就是在做一些基础的服务的搭建，现在微服务这么火，可想而知，...		
©2020 CSDN 皮肤主题: 撸撸猫 设计师:C马雯娟 返回首页		



举报