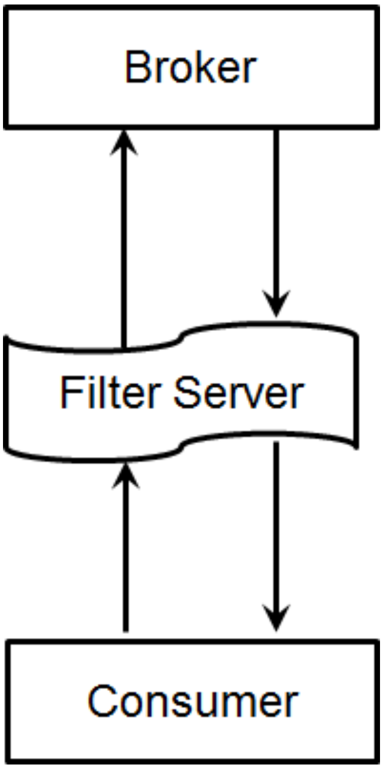
filter_server_guide

Xiaorui Wang edited this page on 29 Mar · 3 revisions

Filter网络架构，以CPU资源换取宝贵的网卡流量资源



启动Broker时，增加以下配置，可以自动加载Filter Server进程

```
filterServerNums=1
```

Filter样本（Consumer仅负责将代码上传到Filter Server，由Filter Server编译后执行）

```
package com.alibaba.rocketmq.example.filter;

import com.alibaba.rocketmq.common.filter.MessageFilter;
import com.alibaba.rocketmq.common.message.MessageExt;


public class MessageFilterImpl implements MessageFilter {

    @Override
    public boolean match(MessageExt msg) {
        String property = msg.getUserProperty("SequenceId");
        if (property != null) {
            int id = Integer.parseInt(property);
            if ((id % 3) == 0 && (id > 10)) {
                return true;
            }
        }

        return false;
    }
}
```

Consumer例子

```
public static void main(String[] args) throws InterruptedException, MQClientEx
    DefaultMQPushConsumer consumer = new DefaultMQPushConsumer("ConsumerGroupN    推模式
    // 使用Java代码，在服务器做消息过滤
    String filterCode = MixAll.file2String("/home/admin/MessageFilterImpl.java
    consumer.subscribe("TopicFilter7", "com.alibaba.rocketmq.example.filter.Me
        filterCode);  订阅主题

    consumer.registerMessageListener(new MessageListenerConcurrently() {
                                  注册消息监听器
        @Override
        public ConsumeConcurrentlyStatus consumeMessage(List<MessageExt> msgs,
                ConsumeConcurrentlyContext context) {
            System.out.println(Thread.currentThread().getName() + " Receive Ne
            return ConsumeConcurrentlyStatus.CONSUME_SUCCESS;
        }
    });

    // Consumer对象在使用之前必须要调用start初始化，初始化一次即可<br>
    consumer.start();
                   启动
    System.out.println("Consumer Started.");
}
```