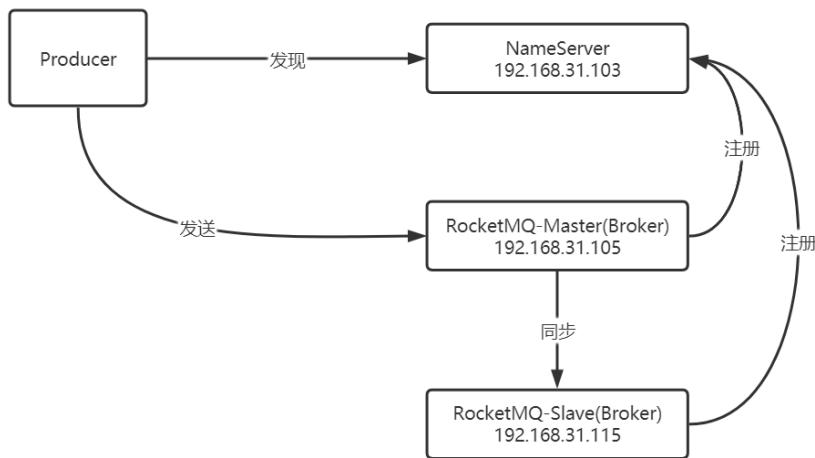


RocketMQ消息发送之普通消息

架构拓扑



- NameServer: 192.168.31.103
- Master: 192.168.31.105
- Slave: 192.168.31.111

执行流程

1. Master与Slave启动向NameServer注册
2. 生产者Producer发送数据前从NameServer获取Master的IP、端口等通信参数
3. 生产者Producer向Master发送消息
4. Master向Slave进行消息同步

代码解析

```
//普通消息类型
@Slf4j
public class MessageType1 {
    public static void main(String[] args) {
        //DefaultMQProducer用于发送非事务消息
        DefaultMQProducer producer = new
DefaultMQProducer("ProducerGroupName");
        //注册NameServer地址
        producer.setNamesrvAddr("192.168.31.103:9876");
        //异步发送失败后Producer自动重试2次
        producer.setRetryTimesWhenSendAsyncFailed(2);
        try {
            //启动生产者实例
            producer.start();
            //消息数据
            String data = "{\"title\":\"X市2021年度第四季度税务汇总数据\"}";
            //消息主题
```

```

        Message message = new Message("tax-data", "2021S4", data.getBytes());
        //发送结果
        SendResult result = producer.send(message);
        log.info("Broker响应: " + result);
    }catch (Exception e){
        e.printStackTrace();
    }finally {
        try {
            //关闭连接
            producer.shutdown();
            log.info("连接已关闭");
        }catch (Exception e){
            e.printStackTrace();
        }
    }
}
}
}

```

运行结果:

23:00:48.741 [main] INFO com.itlaoqi.rocketmq.mtype.MessageType1 - Broker响应:

```

SendResult [
  sendStatus=SEND_OK,
  msgId=7F000001347018B4AAC20A1687DC0000,
  offsetMsgId=C0A81F6900002A9F000000000000A21CB,
  messageQueue=MessageQueue [topic=tax-data, brokerName=broker-a,
  queueId=3],
  queueOffset=2
]

```

- sendStatus: 发送状态, SEND_OK代表成功
- msgId: 消息由RocketMQ分配的全局唯一Id, 由 producer客户端生成, 调用方法MessageClientIDSetter.createUniqID()生成全局唯一的Id
- offsetMsgId: Broker 服务端将消息追加到内存后会返回其物理偏移量, 即在commitlog 文件中的偏移量, 然后会生成一个Id
- messageQueue: 消息队列内容
 - topic: 主题名称
 - brokerName: broker服务器名字, 在RocketMQ xxx.properties配置文件中brokerName项定义
 - queueId: queueId队列Id, 默认会初始化4个 (0-3)
- queueOffset: queueId对应队列逻辑上的位置 (偏移量)