按时间查找某个messagequeue的offset 查找某个messagequeue的最大最小offset 查找某个messagequeue的最早存储时间 根据offsetMsgId查找某一条消息 根据topic+msgId查找某一条消息 根据key+时间范围查找满足条件的消息 Client端nameServer配置 MQConsumerInner ConsumeMessageService Client端ip获取 Client端instanceName设置 从nameServ获取信息的周期 与Broker心跳时间 持久化offset的周期 ConsumeMessageOrderlyS NettyRemoti ngClient PullAPIWrap shConsumerI rrentlyService ervice 用于netty通信 DefaultMQPushConsumer MQClientInstance defaultMQPushConsumerImpl MQClientInstance allocateMessageQueueStrategy rebalanceImpl + new RebalancePushImpl(this); offsetStore checkConfig consumeMessageService Start 1) copySubscription订阅信息放入rebalanceImpl **1**创建NettyC<mark>li</mark>entConfig 2this.client<mark>R</mark>emotingProcessor = new ClientRemot<mark>in</mark>gProcessor(this); 用于处理一切<mark>c</mark>onsumer端接收到的消息 3this.mQCli<mark>en</mark>tAPIImpl = new MQClientAPI<mark>Im</mark>pl(this.clientRemotingProcessor); ltopAddressing = new TopAddressing(MixAll.*getWSAddr*()); topAddr用于从指定的url找到nameSer地址,目的: 动态的改变nameSer地址 2this.remotingClient = new <mark>Ne</mark>ttyRemotingClient(nettyClientConfig, null); 2) this.MQC<mark>li</mark>entInstance 3注册processor,只有一个proc<mark>es</mark>sor:clientRemotingProcessor **4**MQAdmin<mark>Im</mark>pl **5**PullMes<mark>sa</mark>geService **6**Rebalan<mark>ce</mark>Service 7this.defaultMQProducer = new DefaultMQProducer(MixAll.*CLIENT_INNER_PRODUCER_GRO UP*) ;注意<mark>此</mark>处new了另一个实例,与上一层的producer的 不是一个实例 **8**Consume<mark>r\$</mark>tatsManager 3) rebalanceImpl.setConsumerGroup rebalanceImpl.setMessageModel rebalanceImpl.setAllocateMessageQueueStrateg rebalanceImpl.setmQClientFactory 4) this.pullAPIWrapper = new PullAPIWrapper(mQClientFactory,consumerGroup(),isUnitMode()); 5) 创建offsetStore 6)区分是否顺序消费创建consumeMessageService=new ConsumeMessageConcurrentlyService(this, this getMessageListener); Start 7) consumeMessageService.start() Start Start 如果没显示指定nameSe<mark>r,</mark>开启定时任务定时从url中获取nameSer 开启定时任务:从Nam<mark>eS</mark>er更新topic broker路由信息 开启定时任务:清除掉<mark>线</mark>broker、定时向broker发心跳 8) mQClie<mark>nt</mark>Factory.start(); 开启定时任务:持久化<mark>of</mark>fset 开启定时任务:根据消<mark>费</mark>情况调整消费者线程池大小 开始pullMessageServi<mark>ce、</mark>rebalanceService this.defaultMQProducer.getDefaultMQProducerImpl ().start(false)注意此处实例,与上一层的producer的不是一个实例 9) updateTopicSubscribeInfoWhenSubscriptionChanged调用MQClientInstance获 取最新的topic路由信息,MQClientInstance调用this.updateTopicSubscribeInfo 更新topic路由信息

创建topic