**实践部分： Apache RocketMQ开源开发实践**

1. 开发任务介绍（背景及意义）

Apache RocketMQ是由阿里巴巴开源的分布式消息和流数据平台，具备低延迟、高并发、高可用、高可靠，可支撑万亿级数据洪峰的分布式消息中间件。它是企业级互联网架构的核心软件，可以为分布式应用系统提供异步解耦、削峰填谷、流式数据处理、大数据分析的能力，同时具备海量消息处理、高吞吐、可靠重试等互联网应用所需的特性。目前开源Apache RocketMQ已经在金融、电力、物流、游戏、电子商务、共享出行等十几个行业的数十万企业广泛应用，已经成为云计算时代的企业数字化的核心基础设施。

本课程包括理论学习、基础实践、社区贡献三部分。理论部分由企业导师线上讲解Apache RocektMQ的架构原理、核心技术以及使用场景等。基础实践部分需要同学们基于理论部分的学习在阿里云行知实验室（线上）动手实践完成RocektMQ代码编译、集群搭建、demo编写等任务。社区贡献部分主要让同学们体验开源社区的协作与贡献，需要同学们基于前两部分的学习和实践，完整对RocketMQ的代码贡献。通过本课程同学可以理解掌握RocketMQ的核心技术，同时基于对Apache RocketMQ开源社区的实践让同学们了解开源的理念、协作与贡献，引导同学从一名开发中转变成开源贡献者。

1. 任务培养目标
   * 了解分布式消息队列的使用场景。
   * 掌握Apache RocketMQ的架构及原理。
   * 掌握Apache RocketMQ的基础实现。
   * 了解如何接入使用Apache RocketMQ。
   * 提升分布式应用开发能力，提升问题排查定位能力。
   * 掌握如何参与开源社区、如何从开源社区获得帮助。
2. 拟培养人数

8-10人

1. 企业导师介绍：

厉启鹏，开源贡献者， Apache RocketMQ 中国社区发起人 & PMC Member，Linux Open Messaging Advisory Board Member，阿里云技术专家,阿里编程夏令营mentor。之前曾在国家电网电力科学研究院工作，负责国家电网调度控制云基础平台架构设计与落地。目前对分布式中间件、k8s、微服务、物联网、Serverless 感兴趣。

杜恒，Apache RocketMQ PMC Member & Committer，Open Messaging TSC Member，具有多年消息系统、微服务等中间件架构设计及研发经验，对云计算及分布式系统有深刻理解，目前负责 Apache RocketMQ 以及 OpenMessaging 的研发以及标准建设工作。

金融通，Apache RocketMQ PMC Member，Apache RocketMQ 核心贡献者。对分布式系统尤其是分布消息队列有较深理解，作为讲师曾在 Apache RocketMQ 社区和其他开源活动中进行过多次分享。

李伟，Apache RocketMQ北京社区联合发起人, RocketMQ项目Commiter，RocketMQ社区Python客户端项目负责人, Apache Doris项目Contributor, 著有《RocketMQ分布式消息中间件：核心原理与最佳实践》.目前就职于北京某知名在线教育公司大数据平台部, 对消息队列技术、OLAP数据库引擎技术的设计和研发有丰富经验，也热衷于知识分享和社区活动。

1. 课程计划

* 理论学习部分（30%） 3周

*说明：该部分由社区核心贡献者线上授课。*

1. 分布式消息队列基础；
2. Apache RocketMQ功能特性；
3. Apache RocketMQ架构与原理剖析；
4. Apache RocketMQ生态项目；

* 基础实践部分（30%） 3周

*说明：基础实践部分需要同学在阿里云知行实验室动手操作完成；*

1. 基础任务：Apache RocketMQ源代码编译；
2. 基础任务：简单RocketMQ集群搭建1namesrv + 1broker ；
3. 高级任务：完成2m-2s RocketMQ集群搭建
4. 基础任务：消息发送者demo编写：支持普通消息
5. 基础任务：消息发送者demo编写，支持顺序消息
6. 基础任务：消息消费者demo编写：push模式；
7. 基础任务：消息消费者demo编写：完成tag过滤、sql92过滤；
8. 高级任务：消息链路追踪开启与验证；
9. 高级任务：Broker ACL开启与验证；
10. 基础任务：RocketMQ console控制台部署；
11. 高级任务：通过控制台完成操作：重置消费位点
12. 高级任务：RocketMQ Exporter部署和验证；
13. 高级任务：基于Grafana配置RocketMQ监控指标
14. 高级任务：基于Prometheus配置RocketMQ堆积报警

* 开源贡献部分（40%） 4-6周

说明：该部分需要同学参与社区贡献，包括参与社区运营、文档贡献和代码贡献三方面。

1. 社区运营：

订阅社区邮件列表；

社区官微技术文章编辑；

参与社区issue值班；

1. 向社区提交PR贡献技术文档；
2. 向社区提交PR贡献代码，如下题目任选其一：

**题目1:**

在消息领域，消息的送达一般会提供三种语义，即 QoS = 0、1、2，分别对应最多一次，最少一次，有且仅有一次。Apache RocketMQ 作为一款金融级高吞吐、低时延的分布式消息平台，提供了 QoS = 1 的保证， 即严格保证消息不丢， 但是在这种语义保证下，用户需要做严格的幂等处理， 这需要较高的开发成本。

在某些场景下，其对于消息的不丢要求并非那么严格，比如在使用 RocketMQ 来进行RPC 通信的场景，其更希望的是消息不重复，因此希望能够提供 QoS = 0 的语义保证，在这种语义保证中，又有两种情况，即消息级别的 QoS = 0，以及Topic（在Apache RocketMQ 中，所有的消息都是通过broker 中转的方式由发送端投递到消费端到，消息中转的逻辑地址成为 Topic） 级别的 QoS = 0：

消息级别的 QoS = 0，在这种情况下，消息的 QoS 一般是在客户端设置的，该属性与消息相关，不同的客户端发送到同一个topic的消息会有不同的QoS设置，即某些客户端发送QoS = 0 的消息，而其他的可能 QoS = 1，这就意味着同一个Topic 要支持两种投递语义。

Topic级别 QoS = 0，在这种情况下，消息的 QoS 级别一般是在服务端设置的，该属性与Topic 相关，不同的客户端的QoS 不能由客户端决定，而是发往该Topic 的所有消息都有着同样的QoS 设置。

本题目希望同学能为Apache RocketMQ提供QoS = 0 的语义支持，提供最多一次的能力，需求向前兼容。任务包含两部分：

* 完成服务端 QoS = 0 的设计与实现。
* 完成客户端级消息级别的 QoS = 0 的设计与实现

1. 评分标准（考核方式，评分细则）

基础理论、基础实践、社区贡献分别占考核比重的30分、30分、40分。

* 其中基础理论部分需要同学提交一篇关于RocketMQ的技术文章（优秀者将发表到RocketMQ官微）。
* 基础实践部分包括基础任务和高级任务，分别占15分。其中基础任务需要全部完成，未完成基础任务0分。高级任务每个3分，同学可以自己选择。
* 开源贡献部分包括社区运营、贡献文档和贡献代码三部分。其中参与社区运营满分5分、贡献技术文档满分10分，贡献代码满分25分。

1. 课程资源

阿里云知行实验室