

ALIBABA CLOUD

阿里云

消息队列 MQ 快速入门

文档版本：20200921

 阿里云

法律声明

阿里云提醒您阅读或使用本文档之前仔细阅读、充分理解本法律声明各条款的内容。如果您阅读或使用本文档，您的阅读或使用行为将被视为对本声明全部内容的认可。

1. 您应当通过阿里云网站或阿里云提供的其他授权通道下载、获取本文档，且仅能用于自身的合法合规的业务活动。本文档的内容视为阿里云的保密信息，您应当严格遵守保密义务；未经阿里云事先书面同意，您不得向任何第三方披露本手册内容或提供给任何第三方使用。
2. 未经阿里云事先书面许可，任何单位、公司或个人不得擅自摘抄、翻译、复制本文档内容的部分或全部，不得以任何方式或途径进行传播和宣传。
3. 由于产品版本升级、调整或其他原因，本文档内容有可能变更。阿里云保留在没有任何通知或者提示下对本文档的内容进行修改的权利，并在阿里云授权通道中不时发布更新后的用户文档。您应当实时关注用户文档的版本变更并通过阿里云授权渠道下载、获取最新版的用户文档。
4. 本文档仅作为用户使用阿里云产品及服务的参考性指引，阿里云以产品及服务的“现状”、“有缺陷”和“当前功能”的状态提供本文档。阿里云在现有技术的基础上尽最大努力提供相应的介绍及操作指引，但阿里云在此明确声明对本文档内容的准确性、完整性、适用性、可靠性等不作任何明示或暗示的保证。任何单位、公司或个人因为下载、使用或信赖本文档而发生任何差错或经济损失的，阿里云不承担任何法律责任。在任何情况下，阿里云均不对任何间接性、后果性、惩戒性、偶然性、特殊性或刑罚性的损害，包括用户使用或信赖本文档而遭受的利润损失，承担责任（即使阿里云已被告知该等损失的可能性）。
5. 阿里云网站上所有内容，包括但不限于著作、产品、图片、档案、资讯、资料、网站架构、网站画面的安排、网页设计，均由阿里云和/或其关联公司依法拥有其知识产权，包括但不限于商标权、专利权、著作权、商业秘密等。非经阿里云和/或其关联公司书面同意，任何人不得擅自使用、修改、复制、公开传播、改变、散布、发行或公开发表阿里云网站、产品程序或内容。此外，未经阿里云事先书面同意，任何人不得为了任何营销、广告、促销或其他目的使用、公布或复制阿里云的名称（包括但不限于单独为或以组合形式包含“阿里云”、“Aliyun”、“万网”等阿里云和/或其关联公司品牌，上述品牌的附属标志及图案或任何类似公司名称、商号、商标、产品或服务名称、域名、图案标示、标志、标识或通过特定描述使第三方能够识别阿里云和/或其关联公司）。
6. 如若发现本文档存在任何错误，请与阿里云取得直接联系。

通用约定

格式	说明	样例
 危险	该类警示信息将导致系统重大变更甚至故障，或者导致人身伤害等结果。	 危险 重置操作将丢失用户配置数据。
 警告	该类警示信息可能会导致系统重大变更甚至故障，或者导致人身伤害等结果。	 警告 重启操作将导致业务中断，恢复业务时间约十分钟。
 注意	用于警示信息、补充说明等，是用户必须了解的内容。	 注意 权重设置为0，该服务器不会再接受新请求。
 说明	用于补充说明、最佳实践、窍门等，不是用户必须了解的内容。	 说明 您也可以通过按Ctrl+A选中全部文件。
>	多级菜单递进。	单击设置> 网络> 设置网络类型。
粗体	表示按键、菜单、页面名称等UI元素。	在结果确认页面，单击确定。
<code>Courier</code> 字体	命令或代码。	执行 <code>cd /d C:/window</code> 命令，进入Windows系统文件夹。
<i>斜体</i>	表示参数、变量。	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[] 或者 [a b]	表示可选项，至多选择一个。	<code>ipconfig [-all -t]</code>
{ } 或者 {a b}	表示必选项，至多选择一个。	<code>switch {active stand}</code>

目录

- 1.快速入门概述 ----- 05
- 2.快速使用 TCP 协议的 SDK 收发普通消息 ----- 07
 - 2.1. 开通消息队列服务 ----- 07
 - 2.2. 创建资源 ----- 08
 - 2.3. 调用 TCP 协议的 SDK 收发普通消息 ----- 13
- 3.快速使用 HTTP 协议的 SDK 收发普通消息 ----- 15
 - 3.1. 开通消息队列服务 ----- 15
 - 3.2. 创建资源 ----- 16
 - 3.3. 调用 HTTP 协议 SDK 收发普通消息 ----- 21
- 4.快速入门常见问题 ----- 25
 - 4.1. 消息队列 RocketMQ 版是否可以在公网访问？ ----- 25
 - 4.2. 消息队列 RocketMQ 版是否能保证消息不重复？ ----- 26
 - 4.3. 消息队列 RocketMQ 版消息在服务器保存多长时间？ ----- 26
 - 4.4. 消息队列 RocketMQ 版消息 Body 的长度限制是多少？ ----- 26
 - 4.5. 控制台显示的消息堆积量是否包含了 Topic 下所有 Tag 的消息？ ----- 26

1.快速入门概述

消息队列 RocketMQ 版

提供了 TCP 协议和 HTTP 协议的多语言客户端 SDK 以收发不同类型的消息。本文介绍通过两种协议的多语言 SDK 收发普通消息的流程和使用须知。

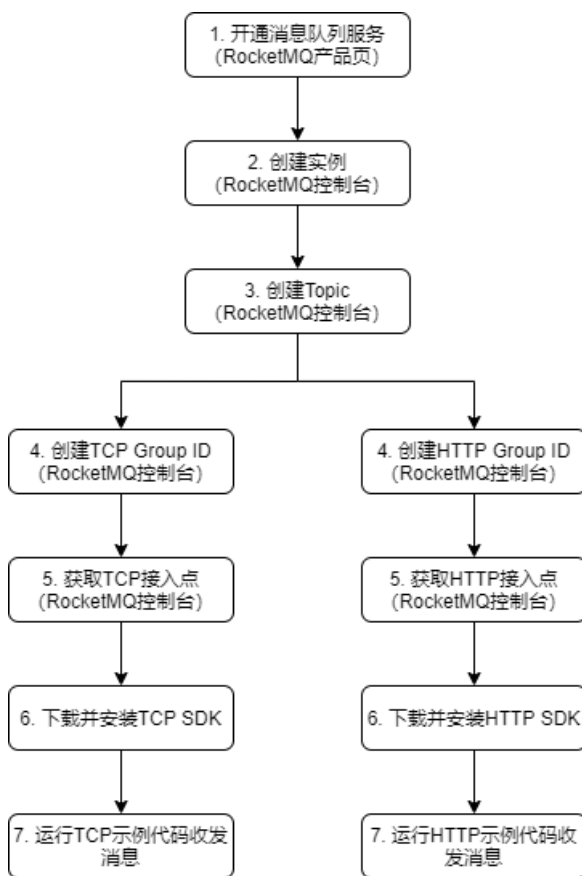
背景信息

消息队列 RocketMQ 版

提供了四种消息类型，详情请参见[消息类型概述](#)。本文以收发普通消息为例进行说明，因此，您按照本文提供的步骤创建的普通消息的 Topic，不能用于收发其他类型的消息。换言之，不同消息类型的 Topic 不能混用。

使用流程

根据您的选择的协议，使用流程如下。



协议使用须知

- 由于消息队列 RocketMQ 版提供的 TCP 协议客户端 SDK 和 HTTP 协议客户端 SDK 不同，因此您所创建的同一个 Group ID 不能混用于 TCP 协议和 HTTP 协议。
- 同一个消息队列 RocketMQ 版实例既有 TCP 协议接入点，又有 HTTP 协议的接入点，您需分别获取 TCP 协议和 HTTP 协议的 SDK 来使用对应协议的接入点，不能混用。
- TCP 协议客户端接入点仅在公网地域有公网接入点，其余地域只提供内网接入点，HTTP 协议在各地域均

提供公网和内网接入点。

- 如果您的应用有跨地域使用消息队列 RocketMQ 版的场景，推荐您使用 HTTP 协议。
- HTTP 协议的 SDK 所支持的功能通过不断迭代，将与 TCP 协议的 SDK 所支持的功能逐渐对齐。HTTP 协议 SDK 的使用须知请参见 [HTTP SDK 版本说明](#)。

后续步骤

- [使用 TCP 协议的 SDK 收发消息](#)
- [使用 HTTP 协议的 SDK 收发消息](#)

2.快速使用 TCP 协议的 SDK 收发普通消息

2.1. 开通消息队列服务

在阿里云官方网站开通消息队列服务后方可开始使用消息队列 RocketMQ 版

。

前提条件

您已注册阿里云账号并完成实名认证。详情请参见[阿里云账号注册流程](#)。

操作步骤

1. 打开[消息队列 RocketMQ 版产品页](#)。
2. 在页面右上角单击登录。
3. 在登录页面输入您的阿里云账号和密码，并单击登录。
4. 在产品页，单击[点击免费开通](#)。
页面跳转至消息队列 RocketMQ 版控制台。
5. 在弹出的提示对话框中，单击[开通MQ服务>>](#)。



6. 在确认订单页面，阅读订单内容以及服务协议后，选中我已阅读并同意消息队列 MQ 服务协议，然后单击立即开通。

确认订单

消息队列 MQ

【超值资源包】：首购 6.8 元试用活动，火热进行中！[购买链接 >>](#)

基本配置	类型	消息队列 MQ
	【超值资源包】：首购 6.8 元试用活动火热进行中！超值资源包链接 >> 【按量付费】：立即购买不会产生任何费用，按实际使用量进行后付费。	
	API 请求次数	API 请求次数费用

按照发送消息 API 请求次数与订阅消息 API 请求次数以及长轮询 API 请求次数进行阶梯计费。
[MQ计费详细说明链接 >>](#)

购买量	TOPIC资源占用	TOPIC 资源占用费
	按照 Topic 的个数进行阶梯计费。 MQ计费详细说明链接 >>	

购买数量

我已阅读并同意 [消息队列 MQ 服务协议](#)

立即开通

当您看到以下内容，则说明您已成功开通消息队列服务。

 **恭喜，开通成功！**

您订购的消息队列（MQ）服务正在开通中，需要 1-5 分钟，请耐心等待！

[管理控制台](#) **0元首购，新用户免费试用活动，火热进行中！**

后续步骤

您可单击[管理控制台](#)继续创建资源。详情请参见[创建资源](#)。

2.2. 创建资源

在调用 TCP 协议的 SDK 收发消息前，您需在消息队列 RocketMQ 版控制台创建相关资源。您在调用 SDK 时需填写这些资源信息。

前提条件

开通消息队列服务

背景信息

针对 TCP 和 HTTP 两种通信协议，

消息队列 RocketMQ 版

分别提供了多语言客户端 SDK 来收发消息。您需创建

消息队列 RocketMQ 版

实例、Topic 和 Group ID。由于 TCP 协议和 HTTP 协议的客户端各不相同，您需为两种协议分别创建不同的 Group ID，换言之，Group ID 不可跨协议使用。

网络访问限制

在使用

消息队列 RocketMQ 版

时，请注意以下网络访问限制：

- 只有在同一个地域下的同一个实例中的 Topic 和 Group ID 才能互通，例如，某 Topic 创建在华东1（杭州）地域的实例 A 中，那么该 Topic 只能被在华东1（杭州）地域的实例 A 中创建的 Group ID 对应的生产端和消费端访问。
- 如果您需要通过公网（例如在本地非阿里云 ECS 服务器）使用消息队列 RocketMQ 版服务，请将 Topic 和 Group ID 都创建在公网地域下的实例中。



生产端和消费端可以部署在本地或者部署在任意地域的 ECS 上，前提是本地服务器或者相应的 ECS 需要能够访问公网。

有关地域的详细介绍请参见[地域和可用区](#)。

创建实例

实例是用于

消息队列 RocketMQ 版

服务的虚拟机资源，会存储消息主题 (Topic) 和客户端 ID (Group ID) 信息。

1. 登录

消息队列 RocketMQ 版
控制台。

2. 在顶部菜单栏，选择地域，如华东1（杭州）。

3. 在左侧导航栏，单击实例列表。

4. 在实例列表页面，单击 + 图标创建实例。

5. 在创建实例对话框，选择实例类型，并输入实例名和描述，然后单击确认。

创建实例
✕

! 1. 实例化详情请参见[资源隔离优化](#)。
2. ons-client 客户端必须升级至最新版本 (JAVA: 1.8.0.Final、C++: 1.1.2、.Net: 1.1.3)。详情请参见[版本说明](#)。

*** 实例类型:** 标准版实例 铂金版实例

i 1. 拥有独立命名空间，资源命名确保实例内唯一，不同实例之间逻辑隔离。
2. 每个地域最多创建 8 个实例，一旦创建成功，所处地域将无法修改。

地域: 🇨🇳 华东1 (杭州) ▾

*** 实例名:** 0/64

长度限制为 3 ~ 64 个字符，可以是中文、英文、数字、短横线 (-) 以及下划线 ()。

*** 描述:** 0/128

确认
取消

标准版和铂金版实例计费详情请参见[规格类型](#)。

您可以在实例列表查看到您刚创建的实例。

创建 Topic

Topic 是

消息队列 RocketMQ 版

里对消息的一级归类，例如创建 Topic_Trade 这一 Topic 来识别交易类消息，消息生产者将消息发送到 Topic_Trade，而消息消费者则通过订阅该 Topic 来获取和消费消息。Topic 的使用说明如下：

- Topic 不能跨实例使用，例如在实例 A 中创建的 Topic A 不能在实例 B 中使用。
- 在同一实例中 Topic 名称必须唯一。
- 您可创建不同的 Topic 来发送不同类型的消息，例如用 Topic A 发送普通消息，Topic B 发送事务消息，Topic C 发送定时和延时消息。

1. 在刚创建的实例所在页面的左侧导航栏，选择 **Topic 管理 > 创建 Topic**。
2. 在创建 Topic 对话框中的 Topic 文本框中，输入 Topic 名称，选择该 Topic 的消息类型为普通消息，输入该 Topic 的描述，然后单击确认。

创建 Topic
✕

i 1. 所有地域（除公网地域外）默认提供内网访问方式，安全、稳定、可靠。
 2. Topic资源占用费请参见[计费详情](#)。不再使用的Topic请及时删除，避免产生不必要的费用。

*** Topic:** 0/64

1. "CID"和"GID"是Group ID的保留字段，Topic命名不能以"CID"和"GID"开头。
 2. Topic只能包含英文、数字、短横线 (-) 和下划线 (_)。
 3. 长度限制在3~64个字符之间。

*** 消息类型:** 普通消息 ▾

普通消息适用于系统间异步解耦、削峰填谷、日志服务、大规模机器的Cache同步以及实时计算分析等场景。

*** 描述:** 0/128

确认
取消

? **说明** 本文以收发普通消息为例进行说明，因此，您按照本文提供的步骤创建的普通消息的 Topic，不能用于收发其他类型的消息。换言之，不同消息类型的 Topic 不能混用。消息类型的更多信息，请参见[消息类型](#)。

您创建的 Topic 将出现在 Topic 列表中。

创建 Group ID

创建完实例和 Topic 后，您需要为消息的消费者（或生产者）创建客户端 ID，即 Group ID 作为标识。Group ID 的使用说明如下：

- 在同一实例中 Group ID 必须唯一。
- Group ID 和 Topic 的关系是 N:N，即一个消费者可以订阅多个 Topic，同一个 Topic 也可以被多个消费者订阅；一个生产者可以向多个 Topic 发送消息，同一个 Topic 也可以接收来自多个生产者的消息。

? **说明** 消费者必须有对应的 Group ID，生产者不做强制要求。

1. 在刚创建的实例所在页面的左侧导航栏，单击 **Group 管理**。
2. 在 **Group 管理** 页面，选择 **TCP 协议 > 创建 Group ID**。

3. 在创建 Group ID 对话框中，输入 Group ID 和描述，然后单击确认。



您创建的 Group ID 将出现在 Group ID 列表中。

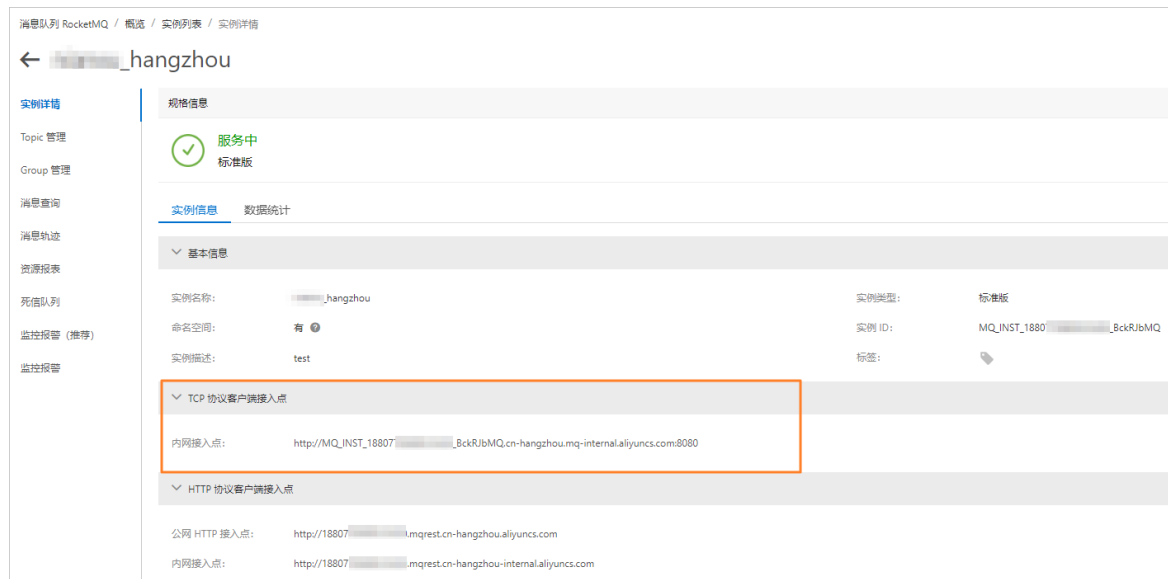
获取接入点

在控制台创建好资源后，您需通过

消息队列 RocketMQ 版

控制台获取实例的接入点。在收发消息时，您需要为生产端和消费端配置该接入点，以此接入某个具体实例或地域的服务。

1. 在刚创建的实例所在页面的左侧导航栏，单击实例详情。
2. 在默认显示的实例信息页签，找到 TCP 协议客户端接入点区域，将鼠标移动至所需的 TCP 协议客户端接入点并单击该接入点即可完成复制。



接入点说明如下：

- 您在控制台看到的 TCP 协议接入点是某地域下某个具体实例的接入点。同一地域下的不同实例的接

入点各不相同。

- 只有在公网地域的实例才有 TCP 协议的公网接入点。其余地域的实例均只有 TCP 协议的内网接入点。
- TCP 协议的接入点不可跨地域使用，详情请参见[配置 TCP 内网接入点](#)。

 **说明** HTTP 协议和 TCP 协议的接入点不可混用。

完成以上准备工作后，您就可以运行示例代码，用

消息队列 RocketMQ 版
进行消息发送和订阅了。

后续步骤


[调用 TCP 协议的 SDK 收发普通消息](#)

2.3. 调用 TCP 协议的 SDK 收发普通消息

在控制台创建完所有资源之后，您可以调用
消息队列 RocketMQ 版
的 TCP 协议的 SDK 收发普通消息。

前提条件

- [创建资源](#)

 **说明** 本文以普通消息为例进行说明，因此您创建的普通消息的 Topic 不能用来收发其他类型的消息，包括定时、延时、顺序和事务消息。换言之，切勿混用不同消息类型的 Topic。

- [创建 AccessKey](#)

下载并安装 TCP 协议的 SDK

推荐您使用

消息队列 RocketMQ 版
提供的以下多语言 TCP 协议的 SDK，请按需获取相应语言的客户端 SDK。
如需使用 TCP 协议其他语言的客户端 SDK，请参见[TCP 协议（社区版）](#)。

调用 TCP 协议的 SDK 发送普通消息

获取相应语言的客户端 SDK 后，您即可运行以下示例代码发送普通消息。

同时，您也可以

[消息队列 RocketMQ 版](#)
[控制台 Topic 管理页面](#)，找到您创建的 Topic，在其操作列，单击发送消息。

Java

.NET

C/C++

查看消息是否发送成功

消息发送后，您可以在

[消息队列 RocketMQ 版](#)
[控制台](#)查看消息发送状态，步骤如下：


1. 在实例所在页面的左侧导航栏，选择消息查询 > 按 Topic 查询。

2. 在搜索框中输入消息发往的 Topic，单击搜索查询消息发送状态。

存储时间表示

消息队列 RocketMQ 版

服务端存储这条消息的时间。如果查询到此消息，表示消息已经成功发送到服务端。

 **注意** 此步骤演示的是第一次使用

消息队列 RocketMQ 版

的场景，此时消费者从未启动过，所以消息状态显示暂无消费数据。要启动消费者并进行消息订阅请继续下一步操作订阅消息。更多消息状态请参见[消息查询](#)和[查询消息轨迹](#)。

调用 TCP 协议的 SDK 订阅普通消息

消息发送成功后，需要启动消费者来订阅普通消息。请按需运行对应语言的示例代码来启动消费者，并测试订阅消息的功能。请按照说明正确设置相关参数。

Java

.NET

C/C++

查看消息是否订阅成功

完成上述步骤后，您可以在

[消息队列 RocketMQ 版](#)

[控制台](#)查看消费者是否启动成功，即消息是否订阅成功。

1. 在 [消息队列 RocketMQ 版](#) [控制台](#)左侧导航栏，选择 **Group 管理 > TCP 协议**。
2. 找到要查看的 Group ID，单击该 Group ID 所在行操作列的订阅关系。是否在线显示为是，且订阅关系一致，则说明订阅成功。

3.快速使用 HTTP 协议的 SDK 收发普通消息

3.1. 开通消息队列服务

在阿里云官方网站开通消息队列服务后方可开始使用消息队列 RocketMQ 版

。

前提条件

您已注册阿里云账号并完成实名认证。详情请参见[阿里云账号注册流程](#)。

操作步骤

1. 打开[消息队列 RocketMQ 版产品页](#)。
2. 在页面右上角单击登录。
3. 在登录页面输入您的阿里云账号和密码，并单击登录。
4. 在产品页，单击[点击免费开通](#)。
页面跳转至消息队列 RocketMQ 版控制台。
5. 在弹出的提示对话框中，单击[开通MQ服务>>](#)。



6. 在确认订单页面，阅读订单内容以及服务协议后，选中我已阅读并同意消息队列 MQ 服务协议，然后单击立即开通。

确认订单

消息队列 MQ

【超值资源包】：首购 6.8 元试用活动，火热进行中！[购买链接 >>](#)

基本配置	类型	消息队列 MQ
	【超值资源包】：首购 6.8 元试用活动火热进行中！超值资源包链接 >> 【按量付费】：立即购买不会产生任何费用，按实际使用量进行后付费。	
	API 请求次数	API 请求次数费用 按照发送消息 API 请求次数与订阅消息 API 请求次数以及长轮询 API 请求次数进行阶梯计费。 MQ计费详细说明链接 >>
TOPIC资源占用	TOPIC 资源占用费 按照 Topic 的个数进行阶梯计费。 MQ计费详细说明链接 >>	

购买量

购买数量

我已阅读并同意 [消息队列 MQ 服务协议](#)

立即开通

当您看到以下内容，则说明您已成功开通消息队列服务。

 **恭喜，开通成功！**

您订购的消息队列（MQ）服务正在开通中，需要 1-5 分钟，请耐心等待！

[管理控制台](#) **0元首购，新用户免费试用活动，火热进行中！**

后续步骤

您可单击[管理控制台](#)继续创建资源。详情请参见[创建资源](#)。

3.2. 创建资源

在调用 HTTP 协议的 SDK 收发消息前，您需在消息队列 RocketMQ 版控制台创建相关资源。

前提条件

[开通消息队列服务](#)

背景信息

针对 TCP 和 HTTP 两种通信协议，

消息队列 RocketMQ 版

分别提供了多语言客户端 SDK 来收发消息。您需创建

消息队列 RocketMQ 版

实例、Topic 和 Group ID。由于 TCP 协议和 HTTP 协议的客户端各不相同，您需为两种协议分别创建不同的 Group ID，换言之，Group ID 不可跨协议使用。

创建实例

实例是用于

消息队列 RocketMQ 版

服务的虚拟机资源，会存储消息主题（Topic）和客户端 ID（Group ID）信息。

1. 登录

[消息队列 RocketMQ 版控制台](#)。

2. 在顶部菜单栏，选择地域，如华东1（杭州）。

3. 在左侧导航栏，单击实例列表。

4. 在实例列表页面，单击 + 图标创建实例。

5. 在创建实例对话框，选择实例类型，并输入实例名和描述，然后单击确认。

创建实例
✕

! 1. 实例化详情请参见[资源隔离优化](#)。
2. ons-client 客户端必须升级至最新版本 (JAVA: 1.8.0.Final、C++: 1.1.2、.Net: 1.1.3) 。详情请参见[版本说明](#)。

*** 实例类型:** 标准版实例 铂金版实例

i 1. 拥有独立命名空间，资源命名确保实例内唯一，不同实例之间逻辑隔离。
2. 每个地域最多创建 8 个实例，一旦创建成功，所处地域将无法修改。

地域: 🇨🇳 华东1 (杭州) ▾

*** 实例名:** 0/64

长度限制为 3 ~ 64 个字符，可以是中文、英文、数字、短横线 (-) 以及下划线 () 。

*** 描述:** 0/128

确认
取消

标准版和铂金版实例计费详情请参见[规格类型](#)。
您可以在实例列表查看到您刚创建的实例。

创建 Topic

Topic 是

消息队列 RocketMQ 版

里对消息的一级归类，例如创建 Topic_Trade 这一 Topic 来识别交易类消息，消息生产者将消息发送到 Topic_Trade，而消息消费者则通过订阅该 Topic 来获取和消费消息。Topic 的使用说明如下：

- Topic 不能跨实例使用，例如在实例 A 中创建的 Topic A 不能在实例 B 中使用。
- 在同一实例中 Topic 名称必须唯一。
- 您可创建不同的 Topic 来发送不同类型的消息，例如用 Topic A 发送普通消息，Topic B 发送事务消息，Topic C 发送定时和延时消息。

1. 在刚创建的实例所在页面的左侧导航栏，选择 **Topic 管理 > 创建 Topic** 。
2. 在创建 Topic 对话框中的 Topic 文本框中，输入 Topic 名称，选择该 Topic 的消息类型为普通消息，输入该 Topic 的描述，然后单击确认。

创建 Topic
✕

i 1. 所有地域（除公网地域外）默认提供内网访问方式，安全、稳定、可靠。
2. Topic资源占用费请参见[计费详情](#)。不再使用的Topic请及时删除，避免产生不必要的费用。

*** Topic:** 0/64

1. "CID"和"GID"是Group ID的保留字段，Topic命名不能以"CID"和"GID"开头。
2. Topic只能包含英文、数字、短横线 (-) 和下划线 (_)。
3. 长度限制在3~64个字符之间。

*** 消息类型:** 普通消息 ▾

普通消息适用于系统间异步解耦、削峰填谷、日志服务、大规模机器的Cache同步以及实时计算分析等场景。

*** 描述:** 0/128

确认
取消

? **说明** 本文以收发普通消息为例进行说明，因此，您按照本文提供的步骤创建的普通消息的 Topic，不能用于收发其他类型的消息。换言之，不同消息类型的 Topic 不能混用。消息类型的更多信息，请参见[消息类型](#)。

您创建的 Topic 将出现在 Topic 列表中。

创建 Group ID

创建完实例和 Topic 后，您需要为消息的消费者（或生产者）创建客户端 ID，即 Group ID 作为标识。Group ID 的使用说明如下：

- 在同一实例中 Group ID 必须唯一。
- Group ID 和 Topic 的关系是 N:N，即一个消费者可以订阅多个 Topic，同一个 Topic 也可以被多个消费者订阅；一个生产者可以向多个 Topic 发送消息，同一个 Topic 也可以接收来自多个生产者的消息。

? **说明** 消费者必须有对应的 Group ID，生产者不做强制要求。

1. 在刚创建的实例所在页面的左侧导航栏，单击 **Group 管理**。
2. 在 **Group 管理** 页面，选择 **HTTP 协议 > 创建 Group ID**。
3. 在创建 Group ID 对话框中，输入 **Group ID** 和 **描述**，然后单击 **确认**。

创建 Group ID
✕

i 1. Group ID 既可用于生产者，标识同一类 Producer 实例 (Producer ID)，又可用于消费者，标识同一类 Consumer 实例 (Consumer ID)；
2. 同一个 Group ID 不可以共用于 TCP 协议和 HTTP 协议，需要分别申请。

*** Group ID:** 4/64

1. 以 "GID_" 或者 "GID-" 开头，只能包含字母、数字、短横线 (-) 和下划线 (_)；
2. 长度限制在 7~64 字符之间；
3. Group ID 一旦创建，则无法修改。

*** 描述:**

确认
取消

您创建的 Group ID 将出现在 Group ID 列表中。

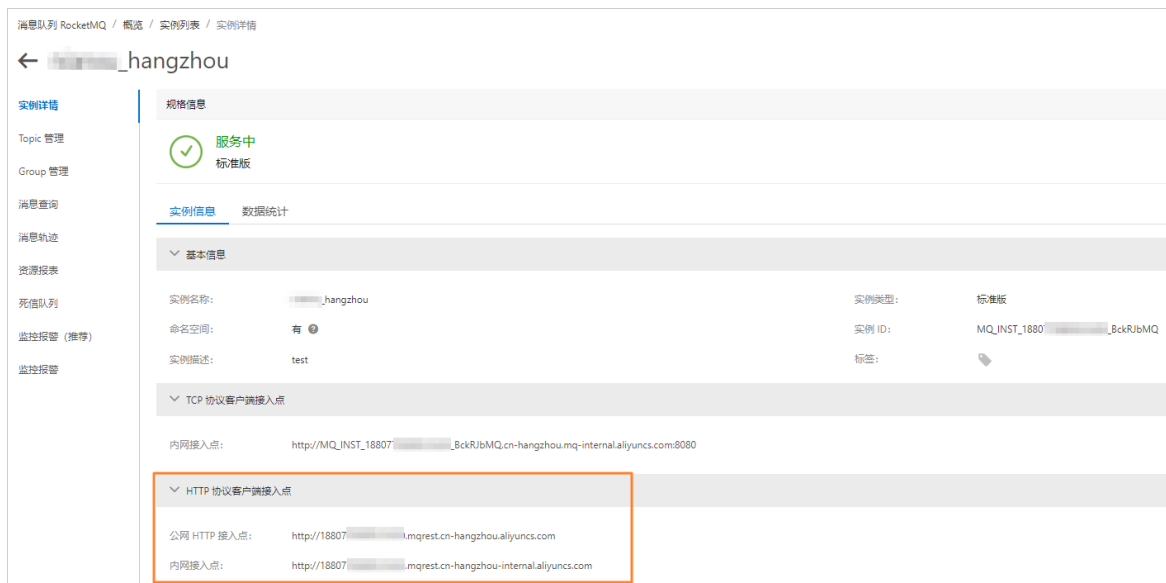
获取接入点

在控制台创建好资源后，您需通过

消息队列 RocketMQ 版

控制台获取实例的接入点。在收发消息时，您需要为生产端和消费端配置该接入点，以此接入某个具体实例或地域的服务。

1. 在刚创建的实例所在页面的左侧导航栏，单击实例详情。
2. 在默认显示的实例信息页签的 HTTP 协议客户端接入点区域，将鼠标移动至所需的 HTTP 协议客户端接入点并单击该接入点即可完成复制。



您在控制台看到的 HTTP 协议接入点是某个地域的接入点，跟具体实例无关。您在收发消息时还需另外设置实例 ID。

? 说明 HTTP 协议和 TCP 协议的接入点不可混用。

后续步骤

调用 HTTP 协议 SDK 收发普通消息

3.3. 调用 HTTP 协议 SDK 收发普通消息

在控制台创建完所有资源之后，您可以调用消息队列 RocketMQ 版的 HTTP 协议的 SDK 收发普通消息。

前提条件





- 创建资源

? 说明 本文以普通消息为例进行说明，因此您创建的普通消息的 Topic 不能用来收发其他类型的消息，包括定时、延时、顺序和事务消息。换言之，切勿混用不同消息类型的 Topic。

- 创建 AccessKey

下载并安装 HTTP 协议的 SDK

消息队列 RocketMQ 版提供以下语言的 HTTP 协议 SDK，请按需下载并安装相应语言的客户端 SDK。

 <p>Java SDK</p> <ul style="list-style-type: none"> • 接入说明 • 版本说明
 <p>PHP SDK</p> <ul style="list-style-type: none"> • 接入说明 • 版本说明
 <p>Go SDK</p> <ul style="list-style-type: none"> • 接入说明 • 版本说明
 <p>Python SDK</p> <ul style="list-style-type: none"> • 接入说明 • 版本说明



调用 HTTP 协议的 SDK 发送普通消息

获取相应语言的客户端 SDK 后，您即可运行以下示例代码发送普通消息。

同时，您还可以在

[消息队列 RocketMQ 版](#)

[控制台 Topic 管理页面](#)，找到您创建的 Topic，在其操作列，单击发送消息。



调用 HTTP 协议的 SDK 消费普通消息

消息发送成功后，需要启动消费者来消费普通消息。请按需运行对应语言的示例代码来启动消费者，并测试消费消息的功能。请按照说明正确设置相关参数。



```

import com.aliyun.mq.http.MQClient;
import com.aliyun.mq.http.MQConsumer;
import com.aliyun.mq.http.common.AckMessageException;
import com.aliyun.mq.http.model.Message;

import java.util.ArrayList;
import java.util.List;

public class Consumer {

    public static void main(String[] args) {
        MQClient mqClient = new MQClient(
            // 设置 HTTP 接入域名。
            "${HTTP_ENDPOINT}",
            // AccessKey ID 阿里云身份验证，在阿里云服务器管理控制台创建。
            "${ACCESS_KEY}",
            // AccessKey Secret 阿里云身份验证，在阿里云服务器管理控制台创建。
            "${SECRET_KEY}"
        );

        // 所属的 Topic。
        final String topic = "${TOPIC}";
        // 您在控制台创建的 Group ID (Consumer ID)。
        final String groupID = "${GROUP_ID}";
    }
}

```

```

final String groupId = "${GROUP_ID}";
// Topic 所属实例 ID, 默认实例为空。
final String instanceId = "${INSTANCE_ID}";

final MQConsumer consumer;
if (instanceId != null && instanceId != "") {
    consumer = mqClient.getConsumer(instanceId, topic, groupId, null);
} else {
    consumer = mqClient.getConsumer(topic, groupId);
}

// 在当前线程循环消费消息, 建议是多开个几个线程并发消费消息。
do {
    List<Message> messages = null;

    try {
        // 长轮询消费消息。
        // 长轮询表示如果 Topic 没有消息则请求会在服务端挂住 3s, 3s 内如果有消息可以消费则立即返回。
        messages = consumer.consumeMessage(
            3, // 一次最多消费 3 条 (最多可设置为 16 条)。
            3 // 长轮询时间 3 秒 (最多可设置为 30 秒)。
        );
    } catch (Throwable e) {
        e.printStackTrace();
        try {
            Thread.sleep(2000);
        } catch (InterruptedException e1) {
            e1.printStackTrace();
        }
    }
    // 没有消息。
    if (messages == null || messages.isEmpty()) {
        System.out.println(Thread.currentThread().getName() + ": no new message, continue!");
        continue;
    }

    // 处理业务逻辑。
    for (Message message : messages) {
        System.out.println("Receive message: " + message);
    }

    // Message.nextConsumeTime 前若不确认消息消费成功, 则消息会重复消费。
    // 消息句柄有时间戳, 同一条消息每次消费拿到的都不一样。
    {
        List<String> handles = new ArrayList<String>();
        for (Message message : messages) {
            handles.add(message.getReceiptHandle());
        }

        try {
            consumer.ackMessage(handles);
        } catch (Throwable e) {
            // 某些消息的句柄可能超时了会导致确认不成功。
            if (e instanceof AckMessageException) {
                AckMessageException errors = (AckMessageException) e;
                System.out.println("Ack message fail, requestId is: " + errors.getRequestId() + ", fail handles:");
                if (errors.getErrorMessages() != null) {
                    for (String errorHandle : errors.getErrorMessages().keySet()) {
                        System.out.println("Handle: " + errorHandle + ", ErrorCode: " +
                            errors.getErrorMessages().get(errorHandle).getErrorCode()
                                + ", ErrorMessage: " +
                            errors.getErrorMessages().get(errorHandle).getErrorMessage());
                    }
                }
            }
        }
    }
}

```

```
    }  
    }  
    continue;  
  }  
  e.printStackTrace();  
}  
}  
} while (true);  
}  
}
```

后续步骤

您可通过查询消息及其轨迹的方式验证消息是否消费成功。详情请参见[消息查询](#)和[查询消息轨迹](#)。

4.快速入门常见问题

4.1. 消息队列 RocketMQ 版是否可以在公网访问?

消息队列 RocketMQ 版支持通过公网访问。

消息队列 RocketMQ 版专门开辟了公网地域 (Region)。在消息队列 RocketMQ 版的控制台创建实例、Topic 以及 Group ID 时，地域选择公网，客户端则可通过消息队列 RocketMQ 版的 TCP 或 HTTP 协议客户端公网接入点访问该 Topic。



接入点说明如下。

地域	TCP 协议客户端接入点	HTTP 协议客户端接入点
公网	只有公网接入点。	只有公网接入点。
除公网外的其他地域	只有内网接入点。	既有公网接入点，又有内网接入点。

公网的网络环境复杂，相对没有内网更稳定。当正式投入生产时，您可选择使用可用性更高的内网地域，部署在阿里云 ECS 上收发消息。

如何快速使用

消息队列 RocketMQ 版

收发消息，请参见以下文档：

- [快速使用 TCP 协议的 Java SDK 收发普通消息](#)
- [快速使用 HTTP 协议的 Java SDK 收发普通消息](#)

4.2. 消息队列 RocketMQ 版是否能保证消息不重复？

绝大多数情况下，消息是不重复的。作为一款分布式消息中间件，在网络抖动、应用处理超时等异常情况下，无法保证消息不重复，但是能保证消息不丢失。

4.3. 消息队列 RocketMQ 版消息在服务器保存多长时间？

存储的消息最多保存 3 天，超过 3 天未消费的消息会被删除。建议配置[监控报警概述](#)实时监控消费进度，并根据报警信息人工介入处理。

4.4. 消息队列 RocketMQ 版消息 Body 的长度限制是多少？

消息大小根据消息类型来限制。具体限制如下所述：

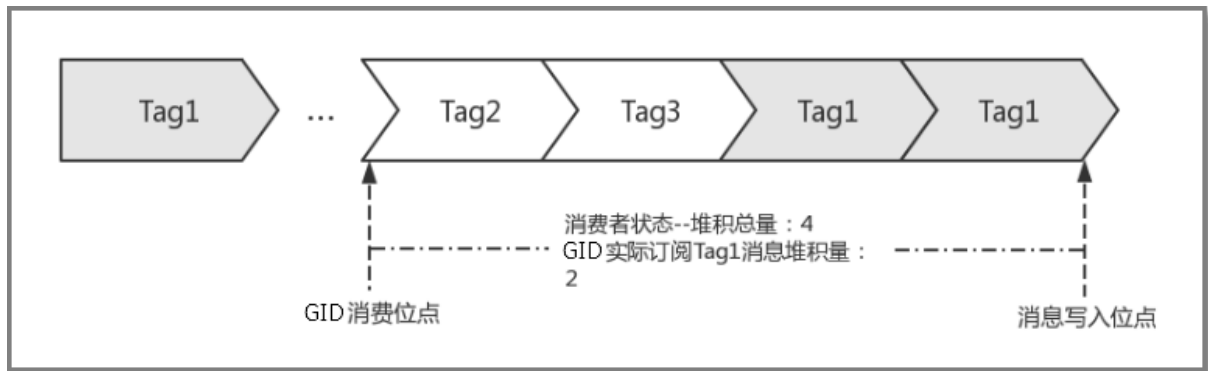
- 普通消息和顺序消息：4 MB
- 事务消息和定时/延时消息：64 KB

4.5. 控制台显示的消息堆积量是否包含了 Topic 下所有 Tag 的消息？

是。

消息生产者将所有类型的 Tag 都发送至同一个 Topic 中，消息按照先后顺序在队列中排列，并维护一个消息写入位点。针对顺序消息，如果某条消息没有被消费，则会阻塞所有消息消费。Group ID 启动时会指明需要订阅的 Tag，并从服务端获取当前的消费位点。服务端从当前 Group ID 的消费位点开始遍历队列中的消息，如果消息的 Tag 符合 Group ID 订阅的 Tag 即投递给 Group ID，不符合则跳过该消息。

如下图所示，Group ID 消费位点往前移动，Tag2、Tag3 的消息会在服务端被过滤掉，Tag1 的消息为 Group ID 所需要的，会投递给 Group ID。



因此您在控制台的消费者状态 > 消息堆积总量 看到的是未被过滤的堆积总量，包含了所有 Tag 的消息量。当订阅方不在线但发送方仍继续发送消息时，堆积数量可能会增长，因为计算规则是发送消息减去已处理的消息。