

消息服务 MNS

产品简介

产品简介

产品概述

阿里云消息服务（Message Service）是一种高效、可靠、安全、便捷、可弹性扩展的分布式消息服务。MNS能够帮助应用开发者在他们应用的分布式组件上自由的传递数据、通知消息，构建松耦合系统。

消息服务同时支持各种类型消息推送，其中和短信前后端的无缝整合更高效的为用户提供了大批量短信发送能力。

消息服务提供了两种模型：

- 队列模型
- 主题模型

两种功能模型的区别：

- 队列模型支持一对一发送和接收消息；
- 主题模型支持一对多发布和订阅消息，并且支持多种消息推送方式。

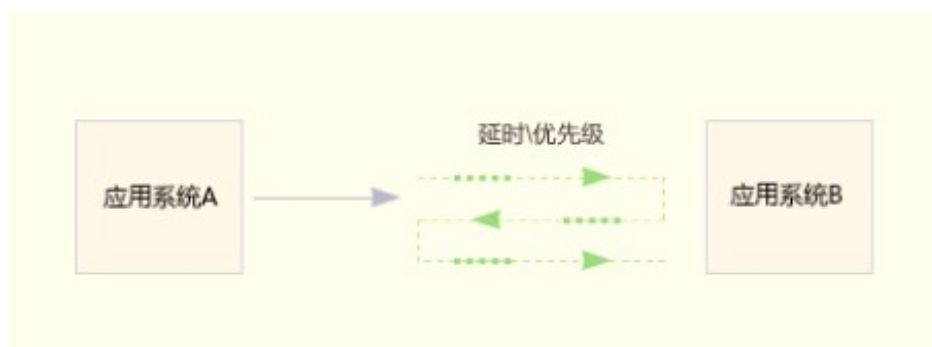
队列模型简介

队列模型旨在提供高可靠高并发的一对一消费模型。即队列中的每一条消息都只能够被某一个消费者进行消费。

类比举例说明

队列模型就如同一家旋转寿司店，寿司店中有多个寿司师傅（生产者）在制作精美的寿司（消息），每一份寿司都是独特的，每位顾客（消费者）同时从传送带上拿取中意的寿司进行食用（消费）。

队列模型示意图



队列模型特性

A. 丰富的队列属性配置

我们提供了丰富的队列属性配置选项，您可以进行队列属性的个性化配置来满足不同的应用场景，支持：普通队列、延迟队列、优先级队列等多种队列模式。

B. 支持海量并发访问

支持多个生产者和消费者并发访问同一个队列，并能确保某条消息在取出之后的特定时间段内，无法被其他消费者获得。可以根据业务需求自由伸缩并发访问数。

C. 消息投递保障

在消息有效期内，确保消息至少能被成功消费一次。接入阿里云账号体系，用户间资源隔离，确保您队列中的消息不会被非法获取。

D. 分布式事务消息

完善的分布式环境下事务消息解决方案。

E. 支持日志管理

可以通过日志管理的方式，查看每一条消息发送、接收和删除的完整生命周期。用户可以通过日志管理，方便的进行问题调查。

详见：日志管理

F. 支持云监控

用户可以通过云监控查看队列情况，并且可以自定义报警项，当队列情况不符合期望时，能够及时知晓。

详见：云监控，云监控控制台

主题模型简介

主题订阅模型旨在提供一对多的发布订阅以及消息通知功能，支持用户实现一站式多种消息通知方式：

- 推送到用户指定 HttpServer

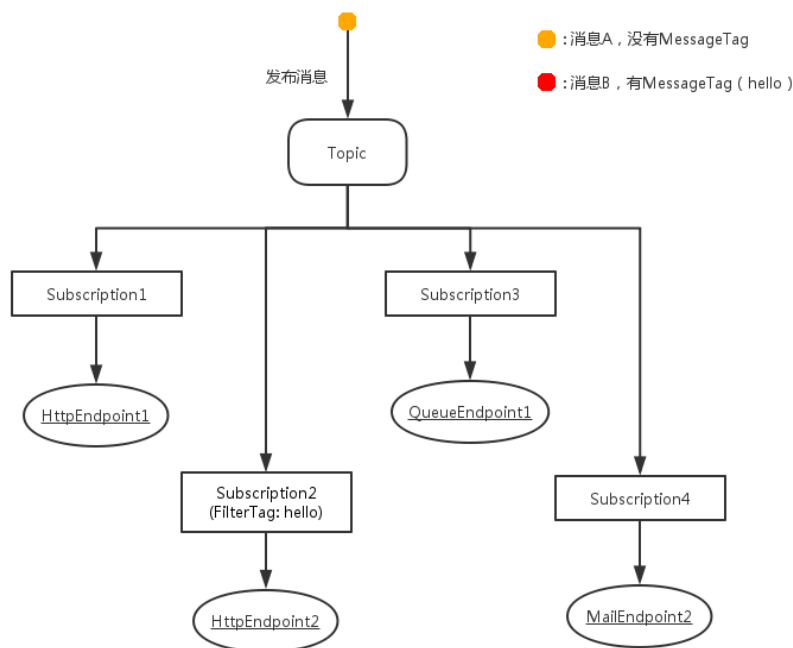
- 推送到用户指定的 Queue（用户可以从该 Queue 拉取消息）
- 推送到邮件（组）
- 推送到短信（列表）
- 移动推送（计划支持）

类比赛例说明

主题就好比一份报纸，多个客户都到邮局订阅了这份报纸，当报纸推出最新一期时，客户（包括邮局的合作伙
伴）可以选择如下方式来获取报纸：

- 让邮局投递员将报纸都投递（推送）到特定的地址（家里）
- 去就近的订阅点（报刊亭）自行获取报纸，（报纸会先被邮局投递员集中送到各个报刊亭）
- 让邮局直接把报纸电子版发送到自己指定的邮箱
- 让邮局把报纸简讯通过短信的方式发送到自己的手机
- 可以让邮局通过WebSock的方式将报纸电子版直接推送到连接的多个终端上
- 可以让报社通过指定移动推送接入点的方式将报纸电子版直接推送到手机APP上

主题模型示意图：



说明：

- 图中所示，Topic上有多个订阅，分别使用了不同的推送方式，有推送到 HttpServer 的，有推送到 Queue 的，有推送到邮箱中的等等。
- 图中所示，一条消息发布到 Topic 中后，会被分别推送到不同的订阅指定的 Endpoint 中。
- 图中所示，订阅支持消息过滤，可以在订阅中指定**过滤标签**。对于没有指定**过滤标签**的订阅，发送消息时，无论有没有指定**消息标签**，消息都可以推送到指定的接收端；对于指定了**过滤标签**的订阅（Subscription2），发送的消息，只有指定了匹配的**消息标签**（红色的消息B），消息才会推送到指

定的接收端。

主题模型特性

- 支持通知消息
- 支持一对多广播消息
- 支持消息标签过滤
- 支持多种投递方式
- 消息投递保障
- 支持云产品事件通知
- 支持日志管理
- 支持云监控

A. 支持通知消息

1) MNS服务端主动将消息推送给用户指定的回调地址（消息通知地址），消除用户端程序不必要的轮询和资源消耗。用户可以将资源集中在处理自身业务的逻辑中。并且也可以避免对其他消息服务产品客户端的依赖，避免因为过多代码引入从而影响自身服务的稳定性。

详见：HttpEndpoint 处理，HttpEndpoint 签名，HttpEndpoint Java，HttpEndpoint 本地 Debug 工具

2) 同时，对于由于网络或者其他原因造成用户不方便提供回调地址的，MNS也支持将消息推送到 Queue 中，用户仍然可以直接到 Queue 中获取消息。原先使用“队列模型”的消费者，可以无缝消费主题中的消息。

详见：Queue 推送，广播拉取消息

B. 支持一对多广播消息

1) 发布到主题中的一条消息可以同时被多个订阅者订阅，会被按照多个订阅指定的推送方式和 Endpoint 地址推送过去。用户同时可以从多个（种）渠道接收到这条消息。

2) 用户仅需发布一次，就可以将消息推送到多个接收端，保证了消息发布的原子性。

C. 支持消息标签过滤

在支持一对多广播消息的基础上，在订阅中还支持对消息进行过滤。可以订阅 Topic 中带有特定标签的消息。在创建订阅时指定消息过滤标签，然后 PublishMessage 时指定消息标签，MNS在推送消息时会根据标签进行过滤，仅推送消息标签与订阅中指定的过滤标签匹配的消息到指定 Endpoint 上。

详见：主题接口规范。

D. 支持多种投递方式

MNS支持多种消息投递方式，包括但不限于：

- 推送到 Http(s)Server，详见：HttpEndpoint 处理，HttpEndpoint 签名，HttpEndpoint Java，HttpEndpoint 本地 Debug 工具。
- 推送到 Queue，详见：Queue 推送，广播拉取消息。
- 推送到邮箱，详见：邮件推送。
- 推送到短信，详见：迁移手册。

- 接入移动推送（计划支持）。

E. 消息投递保障

在消息有效期内，保证发布到 Topic 中的消息会按照指定的策略和格式推送给用户程序。推送策略包括：

退避重试（BACKOFF_RETRY）

重试 3 次，每次重试的间隔时间是 10秒 到 20秒 之间的随机值

指数衰减重试（EXPONENTIAL_DECAY_RETRY），

重试 176 次，每次重试的间隔时间指数递增至 512秒，总计重试时间为1天；每次重试的具体间隔为：1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 512 ... 512 秒 (共167个512)

推送消息格式包括：

- XML：消息体为XML格式，包含消息正文和消息属性；
- JSON：消息体为JSON格式，包含消息正文和消息属性；
- SIMPLIFIED（精简格式，即文本格式）：消息体即用户发布的消息，不包含任何属性信息；

详见：NotifyStrategy，NotifyContentFormat

F. 支持云产品事件通知

MNS提供了事件通知的功能，用户可以通过在MNS上创建事件通知的规则，来关注其他阿里云产品指定资源上产生的事件，并且由MNS以消息的方式主动推送到指定的接收端。目前已接入OSS，用户可以通过事件通知功能监听和处理 OSS 中 Bucket 的管理事件、Bucket 中 Object 的创建、删除、更新等事件。

详见：OSS 事件通知

G. 支持日志管理

可以通过日志管理的方式，查看每一条消息发布进主题的记录，以及每一次推送给用户的记录，包括重试推送的记录、推送结果等。能够展示每条消息的生命周期。用户可以通过日志管理，方便的进行问题调查。

详见：日志管理

H. 支持云监控

可以通过云监控查看消息推送的记录，及时查看和监控消息的推送情况，当推送失败次数比较多时，能够通过添加报警的方式及时知晓。

详见：云监控，云监控控制台

短信发送能力

消息服务同时具备通信服务的能力，支持快速发送短信验证码、短信通知、推广短信。完美支撑双11期间的

2亿用户发送6亿短信。三网合一专属通道，与工信部携号转网平台实时互联。电信级运维保证，时时监控自动切换，到达率高达99%。

短信通知和验证码

大容量高并发：支撑双11期间2亿用户发送6亿短信。3秒可达，三网合一专属通道，与工信部携号转网平台实时互联。变量灵活，支持带入变量，内容灵活，可适应支持各业务场景。

推广短信

支持多种推广内容的短信发放，为提升企业产品增加曝光率提供帮助。业务推广、新产品宣讲、会员关怀等进行短信发送。

批量发送

一对多广播消息，发布到主题中的一条消息可以同时被多个订阅者订阅，会被按照多个订阅指定的推送方式和 Endpoint 地址推送过去。

异步通知

可以后端服务处理完成任务时，回调通知用户。进而减少用户，Web前端和后端服务之间大量不必要的轮询请求。

数据统计

可查看请求量、发送成功量、失败量、等统计数据；通过日期、手机号等维度，查看发送详情。

更多产品使用详情，可参考：

[快速入门手册](#)

[控制台使用帮助](#)

[API使用手册](#)

[SDK使用手册](#)

[MNS最佳实践案例](#)

产品优势

消息服务MNS与自建队列集群的区别

核心优势	使用消息服务MNS	自建队列集群
简单易用	您无需自行搭建消息服务,免运维; 标准HTTP RESTful接口，接入方便; 多种语言SDK支持且不断丰富；	需要采购服务器，初期成本投入高； 需安装和运维，后期成本不断增加； 非HTTP RESTful接口，私有协议，通用性以及安全性偏低；
稳定可靠	消息三份拷贝，可靠性高达99.99999999%； 服务可用性高达99.9%； 特有机制确保Always Writable；	数据单机或主从存储，可靠性差； 单机或小集群服务，可用性无保证； 无法确保Always Writable；
安全防护	多层次安全防护和防DDoS攻击； 多用户隔离机制，每个用户配备独立命名空间； 支持主子账号，支持鉴权和授权； 支持HTTPS，VPC访问；	安全防护能力有限
高可扩展性	队列数量以及队列存储容量可扩展性强； 系统规模自动扩展，对用户完全透明； 全球多地域提供服务； 0启动成本，长期综合成本相比物理机降低50%；	队列数量和消息堆积能力有限； 不具备多地域服务能力； 前期较高服务器成本和日益增长的运维成本；

稳定可靠

消息三份拷贝，可靠性高达99.99999999%；
服务可用性高达99.9%；
特有机制确保Always Writable；

大规模高性能

队列数量以及队列存储容量可扩展性超强；

服务规模自动扩展，并发无上限；

贴心的技术服务

不断丰富的最佳实践案例分享；

7*24小时工单支持；

官方旺旺群（群号：51222373）以及钉钉群技术支持；

可定制化现场支持；

名词解释

队列模型名词解释

队列是消息存储的目的地，队列可以分成普通队列和延时队列两类。

消息包含数据和属性，可以分成普通消息和延时消息，在一个队列中能唯一标识一个消息的有 MessageId 和 ReceiptHandle 两种。

Account

即用户的阿里云帐号ID。

队列所有者

开通 MNS 服务的 Account 通过 CreateQueue 接口创建一个消息队列，这个 Account 就是该队列的所有者，队列所有者拥有此队列的所有操作权限。队列所有者对应的 Account ID 在阿里云官网可以查看。

生产者、消费者

生产者，向 MNS 的消息队列发送消息的角色。
消费者，从 MNS 的消息队列获取消息的角色。

普通队列

如果发送消息时不指定消息延时参数，消息发送到普通队列后可立即被消费。

延时队列

如果发送消息时指定了消息延时参数，发送到延时队列的消息需要经过一定的时间后才能被取到。用户可以通过 `CreateQueue` 和 `SetQueueAttribute` 指定队列的延时时长。

Endpoint

MNS的访问域名地址，格式如下：`http://$AccountId.mns.$Region.aliyuncs.com/`

- `mns.<Region>.aliyuncs.com`：Region 是 MNS服务部署的区域，用户可以根据应用需要选择不同的地域。
- `AccountId`：队列所有者的帐号 ID，调用API请求时需替换成实际的帐号ID。

普通消息、延时消息

- 普通消息是指被发送到队列马上可以被取到的消息；
- 延时消息是指消息发送到队列后需要经过一定的时间后才能被取到的消息，延时时长由 `DelaySeconds` 属性定（请参考 `SendMessage` 相关属性定）。

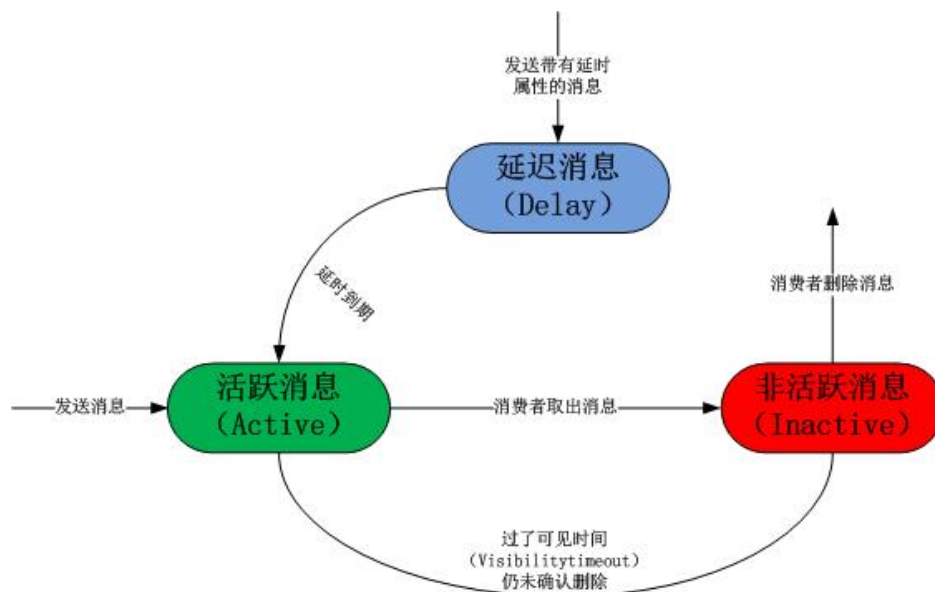
消息ID (MessageID)

`MessageId` 用来标识在队列中的一个消息，在一个队列中每个消息都有唯一的 `MessageId`，但在不同队列之间并不唯一。当消息发送到 MNS 的队列，MNS 会生成一个 `MessageId`，此 ID 一旦产生就不会改变，并在请求响应中返回，用户主要可以使用此 `MessageId` 来做数据校对，但是删除消息必须使用 `ReceiptHandle`。

临时句柄 (ReceiptHandle)

`ReceiptHandle` 是由 MNS 根据当时获取消息的行为产生的临时消息标识，这一点不同于 `MessageId`。当要删除已消费的消息或者改变其 `VisibilityTimeout` 属性值，需要用 `ReceiptHandle` 指定已消费过的消息。`ReceiptHandle` 只能被使用一次，如果 `ReceiptHandle` 标识的消息状态改变其就会失效，消费者只有通过获取消息后重新获取 `ReceiptHandle` 才能进行上述操作。

消息状态 (Message Status)



消息状态转换示意图

普通消息被发送到普通消息队列时，初始状态是 Active，当其被取走后在 VisibilityTimeout 的时间内状态为 Inactive，若超过 VisibilityTimeout 时间后消息还未被删除，消息会重新变成 Active 状态；如果在 VisibilityTimeout 时间内被删除，消息状态为 Deleted。

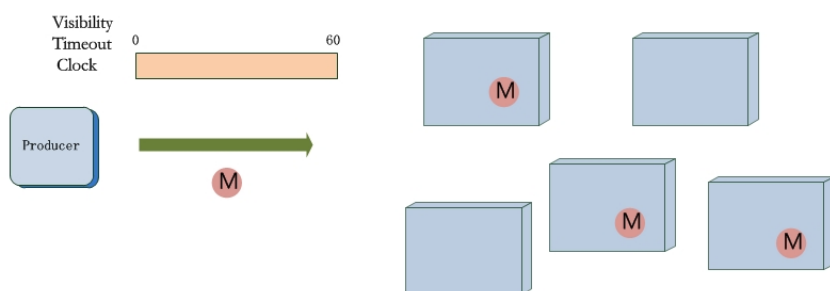
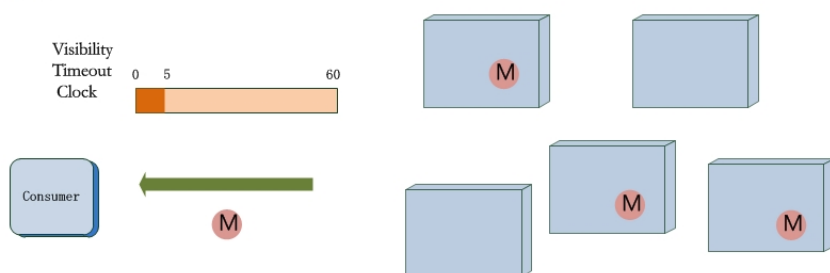
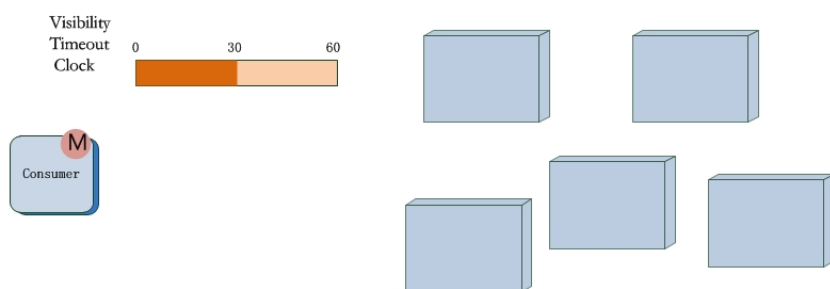
普通消息发送到延时队列时，消息初始状态 Delayed，经过延时队列的 DelaySeconds 属性值设定的时间后，消息状态变成 Active。

延时消息发送到队列（普通队列或者延时队列），消息的初始状态为 Delayed，经过消息的 DelaySeconds 属性指定的时间后消息状态变成 Active。

消息的最长存活时间由创建队列时指定的 MessageRetentionPeriod 属性值决定，超过此时间后消息状态会变成 Expired，将被垃圾回收器回收。

消费者只能取到处于 Active 状态的消息。

消息生命周期

1 生产者发送消息M到队列中**2** 消费者从队列中获取消息M，VisibilityTimeout Clock开始计时**3** 消费者处理消息M后，在Visibility Timeout Clock计时完成前，从队列中删除消息

主题订阅名词解释

发布/订阅模型中的主题，是发布消息的目标和存储地址。

Account

即用户的阿里云帐号ID。

发布者 (Publisher)

消息发布者，往主题发送消息的用户，也可以认为是消息生产者。

订阅者 (Subscriber)

消息订阅者，从主题接收消息的用户，主题所有者默认拥有其主题的订阅权限。

订阅 (Subscription)

订阅关系，消息订阅者只有和主题建立订阅关系才可以接收发布者往该主题发布的消息。

推送地址 (Endpoint)

消息接收端，订阅者用于接收主题消息的地址，例如：http地址，邮箱地址，短信手机号等。

TopicURL

标识 `http://$AccountId.mns.$Region.aliyuncs.com/topics/$TopicName`

`$AccountId` 是用户的帐号 ID

`$Region` 是MNS提供服务的地域，比如：北京、杭州、青岛等

`$TopicName` 是主题的名称

Message

`MessageId` 用来标识在主题中的一个消息，在一个主题中每个消息都有唯一的 `MessageId`，但在不同主题之间并不唯一。

消息发布到 Topic 后，最长存活时间是 1天，过期后消息会变成 Expired 状态，将被垃圾回收器回收。

发展历史

2015/12

通知功能公测，即深圳 region 上线；

控制台支持 RAM 子账号登录，全面支持 RAM 各种授权策；

官方 C++ SDK 发布；

2015/10

性能优化；
官方 PHP 语言 SDK 发布；
美国硅谷 region 上线；

2015/09

队列功能商业化；
官方 C# 语言 SDK；
新加坡 region 上线；

2015/08

通知功能内测，支持媒体转码通知回调功能。

2015/07

MQS 改名 MNS 服务，协议升级；
涵盖通知功能；
支持 RAM 主子账号授权；
提供 Batch 接口；

2014/07

MQS 正式进入阿里云官网，开始公测。

2013/10

支持 OSS 实现异地双集群数据备份功能。

2012/07

MQS 立项，开始第一行代码。

使用限制

消息服务 MNS 对某些具体指标进行了约束和规范，您在使用消息服务 MNS 时注意不要超过相应的限制值，以免程序出现异常。

针对队列模式的限制项和限制值请参见下表。

限制项	限制值	说明
队列数量限制	单个用户 Queue + Topic 数量 ≤ 1000	无
消息大小限制	64 KB	消息大小不得超过该限制，否则消息会被丢弃。
批量接口	支持，batchsize 最大 16 条	无
长轮询	支持，最长 30 秒	无
消息保存时间	15 天	消息最多保留 15 天，超过时间将自动滚动删除，建议按照需求设置。

针对主题模式的限制项和限制值请参见下表。

限制项	限制值	说明
Topic 数量限制	单个用户 Queue + Topic 数量 ≤ 1000	无
单 Topic 订阅数限制	100 个	无
消息保存时间	1 天	消息最多保留 1 天，超过时间将自动滚动删除，建议按照需求设置。