消息队列 RabbitMQ 版

产品简介

产品文档

版权所有：腾讯云计算（北京）有限责任公司

【版权声明】

©2013-2024 腾讯云版权所有

本文档著作权归腾讯云单独所有，未经腾讯云事先书面许可，任何主体不得以任何形式复制、修改、抄袭、传播全部或部分本文档内容。

【商标声明】

及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。

【服务声明】

本文档意在向客户介绍腾讯云全部或部分产品、服务的当时的整体概况，部分产品、服务的内容可能有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

文档目录

产品简介
- 产品概述
- 产品优势
- 应用场景
- 使用限制
- 相关概念
  - 基础概念
  - Exchange

版权所有：腾讯云计算（北京）有限责任公司

产品简介

产品概述

最近更新时间：2024-01-03 11:39:16

消息队列 TDMQ RabbitMQ 版（TDMQ for RabbitMQ，简称 TDMQ RabbitMQ 版）是一款腾讯自主研发的消息队列服务，支持 AMQP 0-9-1 协议，完全兼容开源 RabbitMQ 的各个组件与概念，同时具备计算存储分离，灵活扩缩容的底层优势。

TDMQ RabbitMQ 版拥有极为灵活的路由来适应各类业务的消息投递规则，能有缓冲上游流量压力的能力，保证消息系统的稳定运行。常用于系统间的异步通信和服务解耦，减轻不同服务之间的依赖，广泛应用于金融，政务等行业的分布式系统中。

基础架构

TDMQ RabbitMQ 版的基本概念如下：

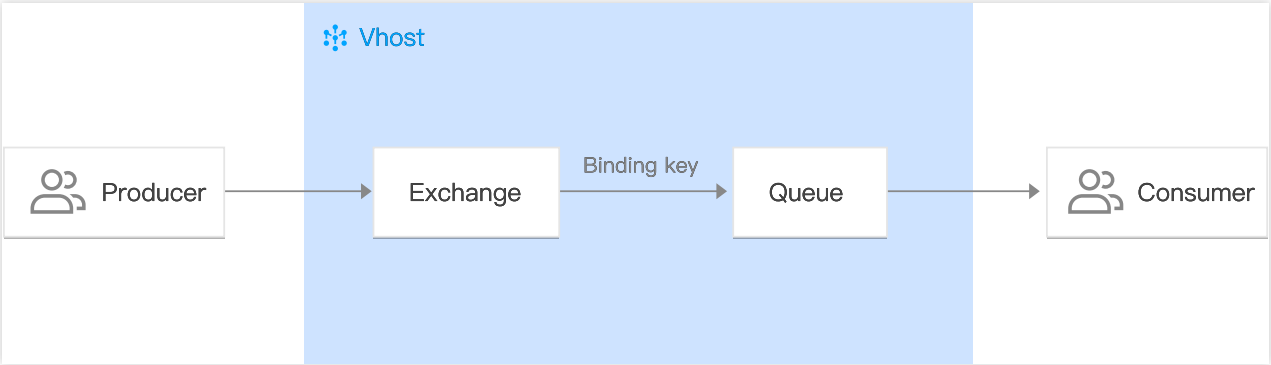
生产者：向 Exchange 发送消息。

Vhost：用作逻辑隔离，不同 Vhost 之间的 Exchange 和 Queue 相互隔离，互不干扰。

Exchange：接收来自生产者的消息并将消息路由到 Queue 的组件。

Queue：存储消息的缓冲区，供消费者消费消息。

消费者：从 Queue 拉取消息进行消费。



更多关于 TDMQ RabbitMQ 版的概念，请参考 相关概念。

产品优势

最近更新时间：2024-01-03 11:39:16

兼容开源

支持 AMQP 0-9-1 版本标准协议，完全支持开源 RabbitMQ 社区和 Queue、Exchange、Vhost 组件。一键迁移开源 RabbitMQ 元数据，实现迁移上云零成本。

功能完备

TDMQ RabbitMQ 版支持原生 RabbitMQ 的各类消息模型。支持死信交换机与备用交换机，用户无需担心由于消息过期、路由失败等因素造成的消息丢失。

稳定可靠

持久化机制确保了 TDMQ RabbitMQ 版的高可靠性。设置 Exchange、Queue、消息的持久化，保证服务重启后元数据与消息内容不丢失。消息采用三副本存储策略，某台物理机故障时，能够实现数据的快速迁移，保证用户数据3个备份可用，服务可用性达99.95%。

高扩展性

TDMQ RabbitMQ 版相比于开源 RabbitMQ 支持更高的队列数量，可扩展能力强，底层系统可根据业务规模自动弹性伸缩、扩容/缩容集群规模，对用户透明。

易用免运维

提供 API 访问接口，支持开源所有语言和版本的 SDK。提供腾讯云平台整套运维服务，实时监控告警，帮助用户快速发现并解决问题，保证服务的可用性。

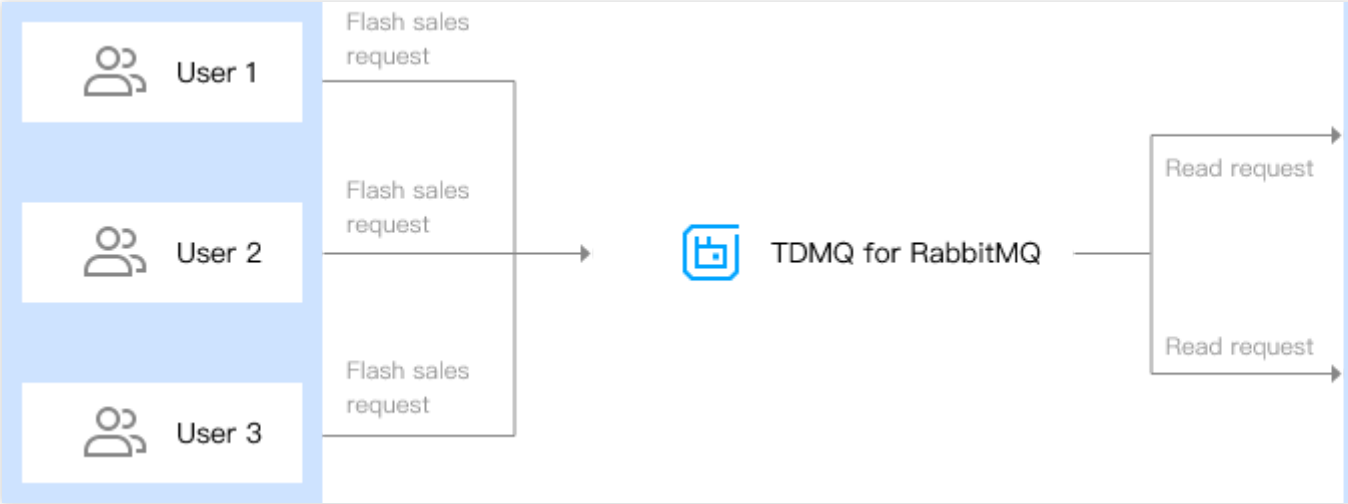
版权所有：腾讯云计算（北京）有限责任公司

应用场景

最近更新时间：2024-01-03 11:39:16

秒杀系统流量削峰

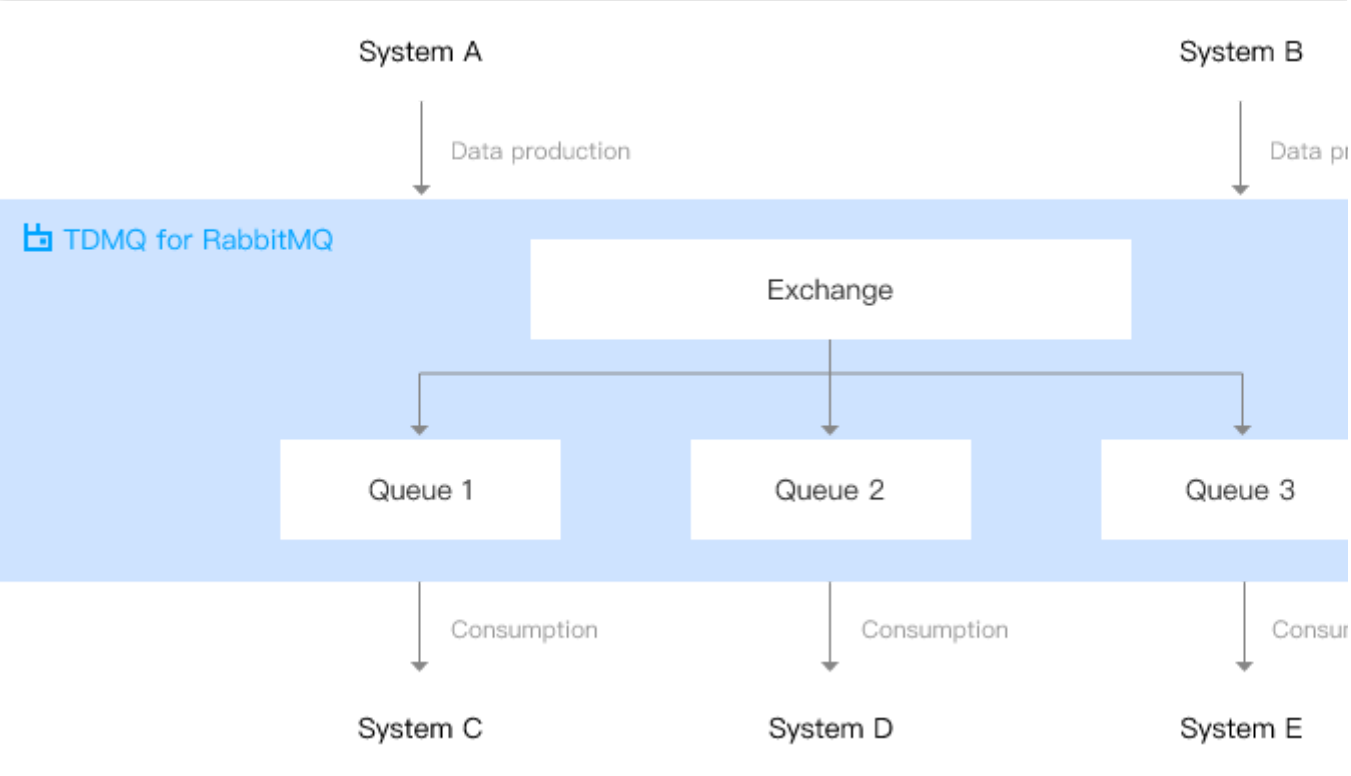
秒杀系统可能因瞬时流量过大导致系统“宕机”，TDMQ RabbitMQ 版缓冲上游的流量压力，保证消息系统的稳定运行。



业务系统异步解耦

交易系统的订单数据涉及下游上百个业务系统，如发货、物流、订单等。TDMQ RabbitMQ 版可以实现系统间的异步通信和服务解耦，减轻不同服务之间的依赖，提升处理效率，保证系统稳定性。

版权所有：腾讯云计算（北京）有限责任公司


System A
System B
Data production
TDMQ for RabbitMQ
Exchange
Queue 1
Queue 2
Queue 3
Consumption
Consumption
System C
System D
System E

使用限制

最近更新时间：2024-06-26 15:56:25

本文列举了 TDMQ RabbitMQ 版专享集群对集群和字符的限制，请您在使用产品时注意不要超出对应的限制值，避免出现异常。

集群

限制类型	限制说明
集群名称长度	3个字符 - 64个字符
集群名称规范	不能为空，只能包含数字、字母、“-”和“_”

Vhost

限制类型	限制说明
Vhost 名称长度	1个字符 - 64个字符
Vhost 名称规范	不能为空，只能包含字母、数字、“.”、“-”及“_”
Vhost 数量	每个集群限制20个

Exchange

限制类型	限制说明
Exchange 名称长度	1个字符 - 64个字符
Exchange 名称规范	不能为空，只能包含字母、数字、“.”、“-”及“_”
Exchange 路由类型	可选项包含direct，fanout，topic及headers，创建时必选其一
Exchange 数量	每个集群限制1000个

Queue

限制类型	限制说明
Queue 名称长度	1个字符 - 64个字符
Queue 名称规范	不能为空，只能包含字母、数字、“.”、“-”及“_”

版权所有：腾讯云计算（北京）有限责任公司

Queue 数量	每个集群限制1000个

Channel

限制类型	限制说明
Connection 通道数	每个 Connection 的最大通道数为1024，如果用户在这个连接上创建的通道数超过这个限制则无法再创建通道

版权所有：腾讯云计算（北京）有限责任公司

相关概念

基础概念

最近更新时间：2024-01-03 11:39:16

本文主要介绍了在使用 TDMQ RabbitMQ 版中常见的名词及解释说明。

概念	解释
Channel	在客户端的每个物理 TCP 连接里，可建立多个 Channel，每个 Channel 代表一个会话任务。
Connection	TCP 连接，生产者或消费者与 TDMQ RabbitMQ 版间的物理 TCP 连接。
Vhost	虚拟主机（Virtual Host，Vhost），用作逻辑隔离，分别管理各自的 Exchange、Queue 和 Binding，使得应用安全的运行在不同的 Vhost 实例上，相互之间不会干扰。一个实例下可以有多个 Vhost，一个 Vhost 里面可以有若干个 Exchange 和 Queue。生产者和消费者连接消息队列 RabbitMQ 版需要指定一个 Vhost。
Queue	Queue（队列）是 RabbitMQ 的内部对象，用于存储消息。每个消息都会被投入到一个或多个 Queue 里。
Exchange	生产者将消息发送到 Exchange，由 Exchange 将消息路由到一个或多个 Queue 中（或者丢弃）。Exchange 根据消息的属性或内容路由消息。
Routing key	生产者在将消息发送到 Exchange 的时候，一般会指定一个 routing key，来指定这个消息的路由规则，而这个 routing key 需要与 Exchange Type 及 binding key 联合使用才能最终生效。 在 Exchange Type 与 binding key 固定的情况下（在正常使用时一般这些内容都是固定配置好的），我们的生产者就可以在发送消息给 Exchange 时，通过指定 routing key 来决定消息流向哪里。
Binding	RabbitMQ 中通过 Binding 将 Exchange 与 Queue 关联起来，这样 RabbitMQ 就知道如何正确地将消息路由到指定的 Queue了。
Binding key	在绑定（Binding）Exchange 与 Queue 的同时，一般会指定一个 binding key；消费者将消息发送给 Exchange 时，一般会指定一个 routing key；当 binding key 与 routing key 相匹配时，消息将会被路由到对应的 Queue 中。 在绑定多个 Queue 到同一个 Exchange 的时候，这些 Binding 允许使用相同的 binding key。binding key 并不是在所有情况下都生效，它依赖于 Exchange Type，例如 fanout 类型的 Exchange 就会无视 binding key，而是将消息路由到所有绑定到该 Exchange 的 Queue。
Exchange Types	RabbitMQ 常用的 Exchange Type 有 fanout、direct、topic 等。 fanout：fanout 类型的 Exchange 会把所有发送到该 Exchange 的消息路由到所有与它绑定的 Queue 中。 Direct：Direct 类型的 Exchange 会把消息路由到那些 binding key 与 routing key 完全匹配的 Queue 中。

	Topic：该类型与 direct 类型相似，Topic 类型 Exchange 支持多条件匹配和模糊匹配，即使用 Routing Key 模式匹配和字符串比较的方式将消息路由至与其绑定的 Queue 中。

版权所有：腾讯云计算（北京）有限责任公司

Exchange

最近更新时间：2024-01-03 11:39:16

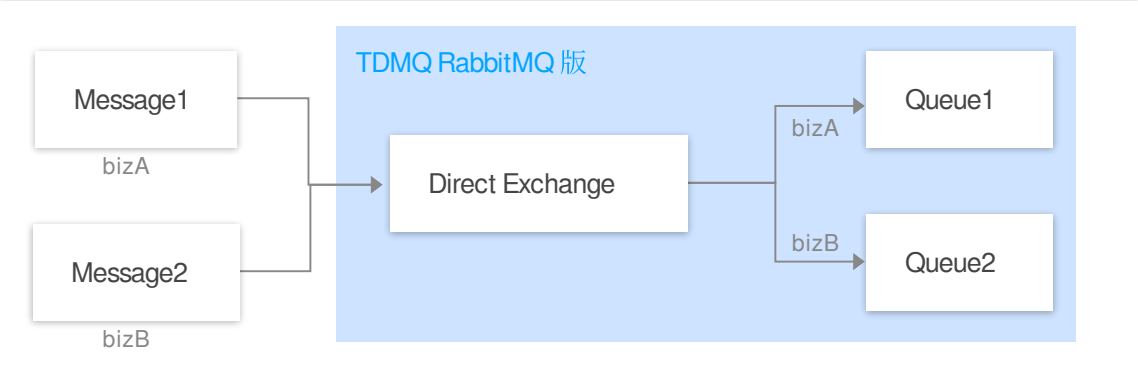
本文介绍 TDMQ RabbitMQ 版中 Exchange 的概念、类型和使用方式。

概念

Exchange 是 TDMQ RabbitMQ 版的消息路由代理，Producer 将消息发送到 Exchange 中，Exchange 根据消息的属性或内容将消息路由到一个或多个 Queue 中（或者丢弃），Consumer 从 Queue 中拉取消息进行消费。
TDMQ RabbitMQ 版目前支持 Direct、Fanout、Topic 和 Header 四种类型的 Exchange。
Direct：该类型 Exchange 会把消息路由到 RoutingKey 和 BindingKey 完全匹配的 Queue 中。
Fanout：该类型 Exchange 会将消息路由到所有与其绑定的 Queue 中。
Topic：该类型 Exchange 支持多条件匹配和模糊匹配，即使用 RoutingKey 模式匹配和字符串比较的方式将消息路由至与其绑定的Queue中。
Header：该类型 Exchange 与 Routing Key 无关，匹配机制是匹配消息中的 Headers 属性信息。在绑定 Queue 与 Headers Exchange 之前声明一个map键值对，通过这个map对象实现消息队列和交换机的绑定。

Direct Exchange

路由规则：Direct Exchange 会把消息路由到 RoutingKey 和 BindingKey 完全匹配的 Queue 中。
应用场景：该类型 Exchange 适用于通过简单字符标识符过滤消息的场景，常用于单播路由。
使用示例：



Message	Routing Key	Binding Key	Queue
Message 1	`bizA`	`bizA`	Queue 1

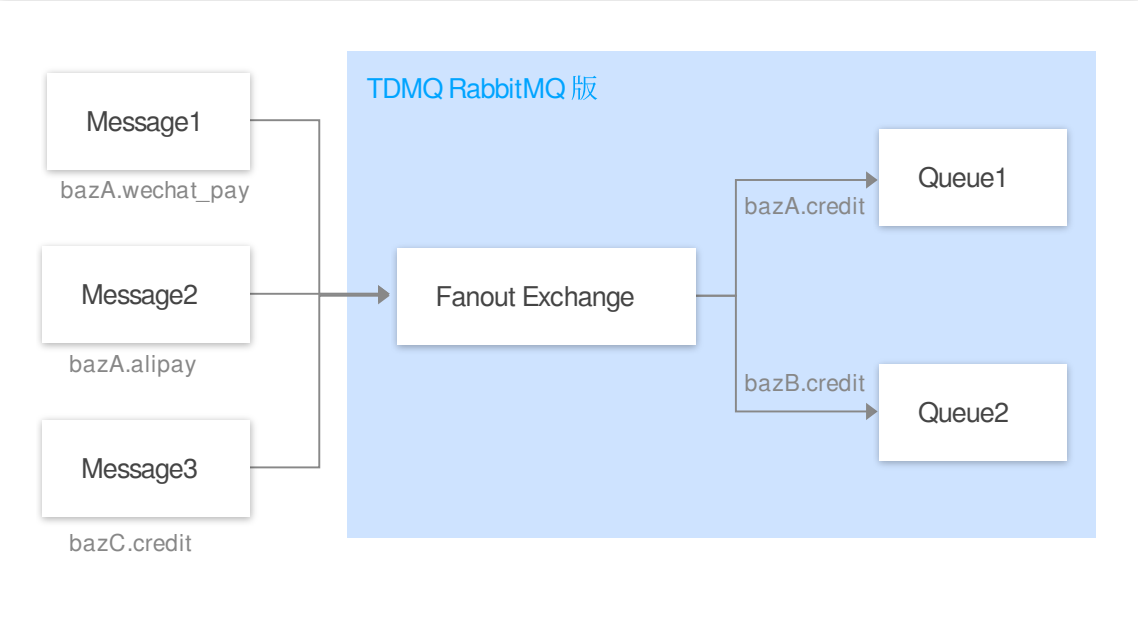
Message 2	`bizB`	`bizB`	Queue 2

Fanout Exchange

路由规则： 该类型 Exchange 会将消息路由到所有与其绑定的 Queue 中。

应用场景：该类型 Exchange 适用于广播消息的场景。例如分发系统使用 Fanout Exchange 来广播各种状态和配置更新。

使用示例



Message	Routing Key	Binding Key	Queue
Message 1	`bazA.wechat_pay`	`bazA.credit` , `bazB.credit`	Queue 1， Queue 2
Message 2	`bazA.alipay`	`bazA.credit` , `bazB.credit`	Queue 1， Queue 2
Message 3	`bazC.credit`	`bazA.credit` , `bazB.credit`	Queue 1， Queue 2

Topic Exchange

路由规则：该类型 Exchange 支持多条件匹配和模糊匹配，即使用 Routing Key 模式匹配和字符串比较的方式将消息路由至与其绑定的 Queue 中。

Topic Exchange 支持的通配符包括星号“*”和井号“#”。

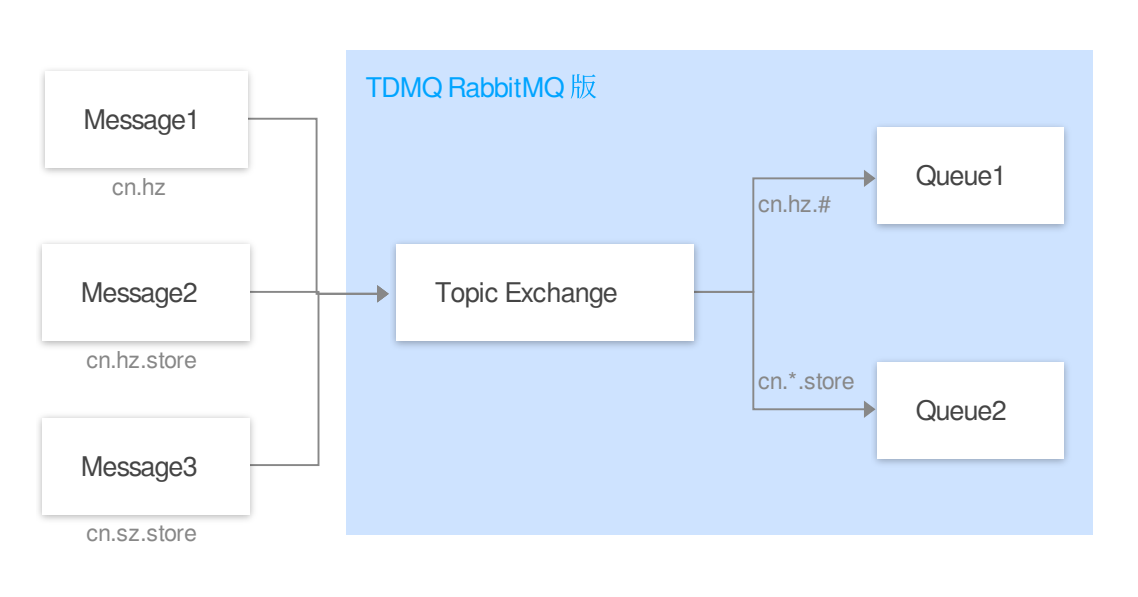
版权所有：腾讯云计算（北京）有限责任公司

星号“*”代表一个英文单词，例如 sh。
井号“#”代表零个、一个或多个英文单词，单词间用英文句号"."分隔，例如 cn.hz。

应用场景

该类型 Exchange 常用于多播路由，例如需要使用 Topic Exchange 分发有关于特定地理位置的数据。

使用示例



Message	Routing Key	Binding Key	Queue
Message 1	`cn.hz`	`cn.hz.#`	Queue 1
Message 2	`cn.hz.store`	`cn.hz.#` 、 `cn.*.store`	Queue 1、Queue 2
Message 3	`cn.sz.store`	`cn.*.store`	Queue 2

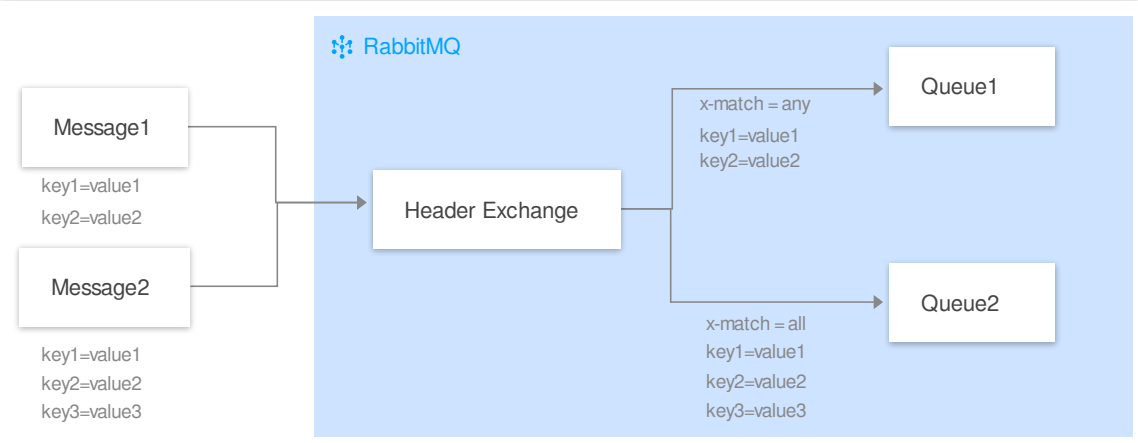
Header Exchange

与 Routing Key 无关，匹配机制是匹配消息中的 Headers 属性信息。在绑定 Queue 与 Headers Exchange 之前声明一个map键值对，通过这个map对象实现Queue 和 Exchange 的绑定。当消息发送到 RabbitMQ 时会取到该消息的 Headers 与 Exchange 绑定时指定的键值对进行匹配；如果完全匹配则消息会路由到该队列，否则不会路由到该队列。

匹配规则x-match有下列两种类型：

x-match = all ：表示所有的键值对都匹配才能接受到消息

x-match = any ：表示只要有键值对匹配就能接受到消息



Message	消息Headers属性	Binding Headers 属性	Queue

Message 1	`key1=value1`
	`key2=value2`
		`x-match = any`
		`key1=value1`
		`key2=value2`
			Queue 1

Message 2	`key1=value1`
	`key2=value2`
	`key3=value3`
		`x-match = any`
		`key1=value1`
		`key2=value2` 、
		
		`x-match = all`
		`key1=value1`
		`key2=value2`
		`key3=value3`
			Queue 1、Queue 2

版权所有：腾讯云计算（北京）有限责任公司