

电脑培训学校	出美国留学机构	MBA报考条件	从零开始学英语	美国留学生中介
陆家嘴写字楼	UI设计培训学校	美甲培训学校	启蒙儿童英语	美容培训学校
前十名装修公司	伤疤如何去掉	日语 家教	雅思留学培训班	办公室 装修
怎么缩小鼻孔	月子中心 费用	自己治疗灰指甲	导航公司	哪里做双眼皮做的好

分类：[java服务端](#) [分布式服务架构](#) [java并发编程](#) [JavaScript生态体系](#) [大数据与人工智能](#)
[数据库](#) [服务器运维](#) [全文搜索](#) [W3C&前端开发](#) [项目构建管理](#) [php](#)
[编程语言](#) [常用软件](#) [软件测试](#)

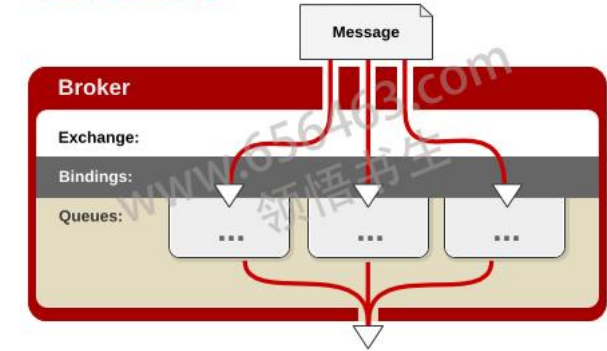
[java教程](#) [freemarker教程](#) [ActiveMQ教程](#) [RabbitMQ 教程](#) [kafka教程](#) [spring教程](#)
[Spring Batch教程](#) [Spring Boot教程](#) [Spring MVC教程](#) [httpClient 教程](#) [Netty教程](#) [shiro教程](#)
[MyBatis 教程](#) [Hibernate教程](#) [JPA教程](#) [OAuth 2.0教程](#) [activiti工作流教程](#) [Quartz教程](#) [Guava教程](#)
[Kotlin教程](#) [Apache Commons教程](#) [WebSocket教程](#) [POI教程](#) [Struts2教程](#) [HTTP教程](#) [TCP/IP教程](#)
[YAML教程](#) [Web Services教程](#) [Memcached教程](#) [JFinal教程](#) [jsp&servlet教程](#) [jstl el 教程](#)
[JSoup教程](#) [JSON教程](#) [tomcat教程](#) [JFreeChart教程](#) [XStream教程](#) [Thymeleaf教程](#) [Groovy教程](#)
[Log4j教程](#) [JasperReports教程](#) [正则表达式](#) [i18n 国际化](#) [JSF教程](#) [设计模式](#) [JAVA GUI教程](#)
[JOGL教程](#) [PDFBox教程](#)

RabbitMQ 三种Exchange(fanout,direct,topic)模式

2019-03-05 23:22 | 来源: 网路

RabbitMQ提供了四种Exchange：fanout,direct,topic,header，header模式在实际使用中较少，本文只对前三种模式进行比较。
性能排序：fanout > direct >> topic。比例大约为11：10：6

1、Fanout Exchange



- 任何发送到Fanout Exchange的消息都会被转发到与该Exchange绑定(Binding)的所有Queue上。
- 1、可以理解为路由表的模式
 - 2、这种模式不需要RouteKey
 - 3、这种模式需要提前将Exchange与Queue进行绑定，一个Exchange可以绑定多个Queue，一个Queue可以同多个Exchange进行绑定。
 - 4、一个发送到交换机的消息都会被转发到与该交换机绑定的所有队列上。很像子网广播，每台子网内的主机都获得了一份复制的消息。
 - 5、如果接受到消息的Exchange没有与任何Queue绑定，则消息会被抛弃。
 - 6、Fanout交换机转发消息是最快的。

```
1 Channel channel = connection.createChannel();
2 channel.exchangeDeclare("exchangeName", "fanout"); //direct fanout topic
3 channel.queueDeclare("queueName");
4 channel.queueBind("queueName", "exchangeName", "routingKey");
5 channel.queueDeclare("queueName1");
6 channel.queueBind("queueName1", "exchangeName", "routingKey1");
7 byte[] messageBodyBytes = "hello world".getBytes();
8 //路由键需要设置为空
9 channel.basicPublish("exchangeName", "", MessageProperties.PERSISTENT_TEXT_PLAIN, messageBodyBytes);
```

2、Direct Exchange

请输入关键词

RabbitMQ 教程

- RabbitMQ进阶教程**
 - RabbitMQ 与Spring整合
 - RabbitMQ 集群搭建
- RabbitMQ入门教程**
 - RabbitMQ 简介
 - RabbitMQ 环境配置-基于linux
 - RabbitMQ 环境配置-基于window
 - RabbitMQ 插件管理
 - RabbitMQ 用户管理
 - RabbitMQ vhost管理
 - RabbitMQ 简单队列
 - RabbitMQ Work模式消息队列
 - RabbitMQ 消息确认方式
 - RabbitMQ 三种Exchange(fanout,direct,topic)模式
 - RabbitMQ 发布订阅模式(Publish、Subscribe)
 - RabbitMQ 路由模式 (Routing) -使用 direct
 - RabbitMQ topic Exchange使用

相关文章

- RabbitMQ topic Exchange使用
- RabbitMQ 路由模式 (Routing) -使用 direct
- RabbitMQ 简介
- RabbitMQ 插件管理
- RabbitMQ 集群搭建
- RabbitMQ 与Spring整合
- RabbitMQ vhost管理
- RabbitMQ 消息确认方式
- RabbitMQ 发布订阅模式(Publish、Subscribe)
- RabbitMQ Work模式消息队列
- RabbitMQ 环境配置-基于windows 10 x64
- RabbitMQ 简单队列
- RabbitMQ 用户管理
- RabbitMQ 环境配置-基于linux
- hibernate三种状态的讲解

最近更新

- 构建Disruptor实例-生产消费模型完成整个入
- 数据生产者向disruptor容器RingBuffer中推
- 数据消费者：实现EventHandler监听接口处理
- Disruptor实现EventFactory接口创建实例对
- 快速了解Sentinel 如何使用
- Sentinel更新计划 Roadmap
- Sentinel快速入门



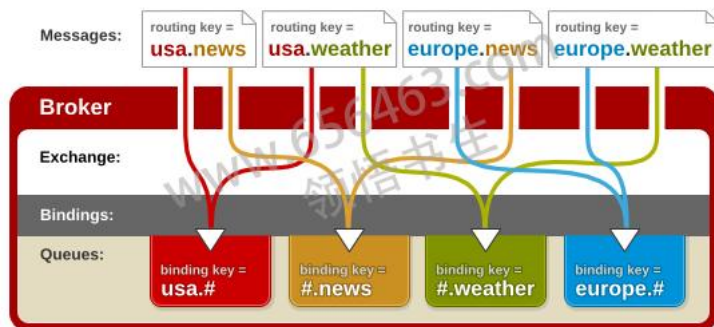
任何发送到Direct Exchange的消息都会被转发到RouteKey中指定的Queue。

- 1、处理路由键
- 2、需要将一个队列绑定到交换机上，要求该消息与一个特定的路由键完全匹配。这是一个完整的匹配。如果一个队列绑定到该交换机上要求路由键 “dog”，则只有被标记为 “dog” 的消息才被转发，不会转发dog.puppy，也不会转发dog.guard，只会转发dog。
- 3、一般情况可以使用rabbitMQ自带的Exchange：“”(该Exchange的名字为空字符串，下文称其为default Exchange)。
- 4、这种模式下不需要将Exchange进行任何绑定(binding)操作
- 5、消息传递时需要一个“RouteKey”，可以简单的理解为要发送到的队列名字。
- 6、如果vhost中不存在RouteKey中指定的队列名，则该消息会被抛弃。

```
1 Channel channel = connection.createChannel();
2 channel.exchangeDeclare("exchangeName", "direct"); //direct fanout topic
3 channel.queueDeclare("queueName");
4 channel.queueBind("queueName", "exchangeName", "routingKey");
5 byte[] messageBodyBytes = "hello world".getBytes();
6 //需要绑定路由键
7 channel.basicPublish("exchangeName", "routingKey", MessageProperties.PERSISTENT_TEXT_PLAIN,
  messageBodyBytes);
```

3、Topic Exchange

Topic Exchange



任何发送到Topic Exchange的消息都会被转发到所有关心RouteKey中指定话题的Queue上

- 1、这种模式较为复杂，简单来说，就是每个队列都有其关心的主题，所有的消息都带有一个“标题” (RouteKey)，Exchange会将消息转发到所有关注主题能与RouteKey模糊匹配的队列。
- 2、这种模式需要RouteKey，也许要提前绑定Exchange与Queue。
- 3、在进行绑定时，要提供一个该队列关心的主题，如“#.log.#”表示该队列关心所有涉及log的消息(一个RouteKey为“MQ.log.error”的消息会被转发到该队列)。
- 4、“#”表示0个或多个关键字，“*”表示一个关键字。如“log.*”能与“log.warn”匹配，无法与“log.warn.timeout”匹配；但是“log.#”能与上述两者匹配。
- 5、同样，如果Exchange没有发现能够与RouteKey匹配的Queue，则会抛弃此消息。

```
1 Channel channel = connection.createChannel();
2 channel.exchangeDeclare("exchangeName", "topic"); //direct fanout topic
3 channel.queueDeclare("queueName");
4 channel.queueBind("queueName", "exchangeName", "routingKey.*");
5 byte[] messageBodyBytes = "hello world".getBytes();
6 channel.basicPublish("exchangeName", "routingKey.one", MessageProperties.PERSISTENT_TEXT_PLAIN,
  messageBodyBytes);
```

书生整理于网络

相关问答

[更多](#)

RabbitMQ RPC跨多个rabbitMQ实例(RabbitMQ RPC across multiple rabbitMQ instances)

看看org.springframework.amqp.rabbit.connection.AbstractRoutingConnectionFactory。它将允许您为不同的虚拟主机或不同的rabbitmq实例创建多个连接工厂。我们将其用于多租户rabbitmq应用程序。Take a look at org.springframework.amqp.rabbit.connection.AbstractRoutingConnectionFactory. It will allow you to ...

RabbitMQ Exchange全部收获(RabbitMQ Exchange with catch all)

specific队列。备用交换将直接将消息传递到generic队列。这可以通过主题来完成，以重用相同的路由键。或者它可以作为扇出交换完成，并且只有一个队列绑定它。 your best option for t ...

使用RabbitMQ与Plone - 芹菜还是不 ? (Using RabbitMQ with Plone - Celery or not?)

首先，问问你自己，如果你需要RabbitMQ的功能，或者只想用Plone在Python中完成一些异步任务。如果你真的不需要RabbitMQ，你可以看看 David Glick 的要点：如何将 Celery 和 Plone 集成在一起（并且仍然使用 RabbitMQ 和 Celery）：
<https://gist.github.com/davisagli/5824662> <https://gist.github.com/davisagli/5824709> 您还可以查看collective.taskqueue（没有C ...

netty和rabbitmq的区别

netty和rabbitmq层次的问题：我知道netty是tcp通信框架，rabbitmq是基于tcp通信封装的一种消息队列。如果包含套节字的话他们之间的关系层次是 socket/nio ---> netty ---> rabbitmq 这种，不知道我理解的有没有错误。 netty和rabbitmq 替换关系：在不考虑数据解析序列化的前提下，单对单的，不涉及延时：能用netty的地方是不是可以用rabbitmq？能用rabbitmq的地方是不是可以使用netty+protobuf替换。

RabbitMQ AMQP队列设计(RabbitMQ AMQP queue design)

回答这个问题：我是否希望所有队列都收到所有消息？如果答案是肯定的，那么你应该使用扇出。如果答案是否定，则应使用直接或主题。直接或主题的要点是队列本身只会根据路由密钥与绑定密钥的匹配来接收消息。队列A应该由消费者C1实例化，并设置为自动删除并且不持久。当C1断开队列时，这种方式将被删除，消息将被丢弃。相反，当交换单独或由生产者进行交换时，队列B和C应该被实例化。应该设置为非自动删除并且可能是持久的。如果您使用的是持久队列，则可能需要持久消息（如果队列A不存在，请不要担心，即使持久消息 ...