The Chat Application using Java RabbitMQ

Mpoki MWAISELA , Ahmad GHALAWINJI

March 18, 2022

1 Introduction

The purpose of this lab is to develop a chat system using Java and based on RabbitMQ library. The system was built on P2P architecture and it could be launched on each host included in the system (no centralized).

2 Design Decision

We use the peer to peer approach, where all nodes in the system are equal and

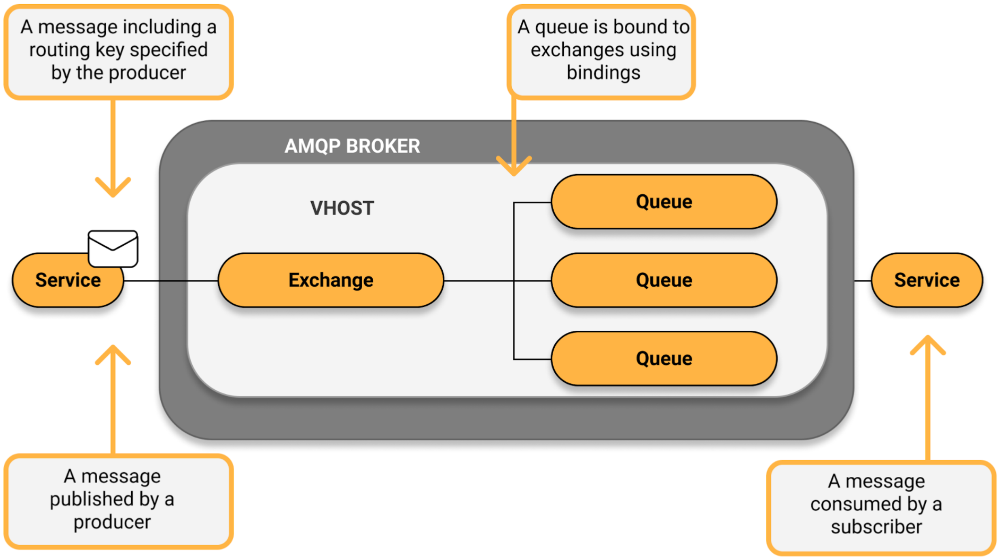
A message including a routing key specified by the producer
A queue is bound to exchanges using bindings
AMQP BROKER
VHOST
Service
Exchange
Queue
Queue
Queue
Service
A message published by a producer
A message consumed by a subscriber

Figure 1: P2P in RabbitMQ.

3 Features

1. Each user can join/leave whenever he/she wants.
2. Each user can join/receive/broadcast whenever he/she wants.
3. Whenever a new client joined, he will receive a copy of all messages.

4 Comparing with Java RMI

- Much simpler and faster implementation.

5 Compile and run

1. **To compile** : javac -cp amqp-client-5.14.2.jar chatApp.java
2. **To run** : java -cp .:amqp-client-5.14.2.jar:slf4j-api-1.7.36.jar:slf4j-simple-1.7.36.jar chatApp NAME
   Note : name refers to the client name used in the chat and should be unique assigned.

6 Unidirectional ring using Java threads

The idea here is that:

1. If I am the sender, I'll just send.
2. If I am not the sender nor the receiver, I'll receive the message from the previous node and forward it toward the next one.
3. If I am the receiver, I'll just receiver the message.

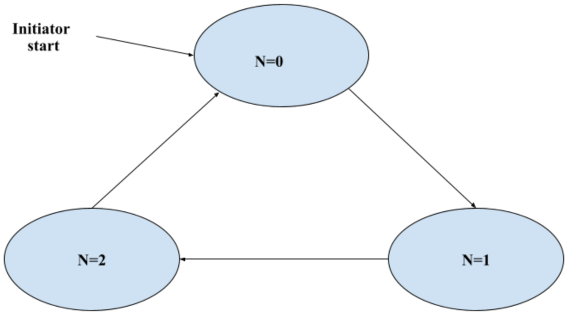

Figure 2: Ring topology with three nodes

The pseudo-code for the algorithm as follow

```
$Bsend(m, source)$
$myid = mynum()$
$numnodes = numofnodes()$
$to = (my_id + 1)modnumnodes$
$from = (myid - 1)modnumnodes$
**if** $(myid == source)$ **then**
    $send(m, to)$
**else**
    **if** $(to == source)$ **then**
        $receive(m, from)$
    **else**
        $receive(m, from)$
        $send(m, to)$
```

Unfortunately, the implemented code didn't work, we didn't have much time to handle it. Hopefully, by the deadline for the final project we will fix the bugs.