RabbitMQ

Message broker



|  |  |
| --- | --- |
| **Date** | 29-03-2021 |
| **Version** | 0.1 |
| **Student Name** | Lars van den Brandt |
| **Student Number** | 434565 |
| **Teacher Fontys** | Tom Meulensteen & Marcel Boelaars |

#### Version

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Changes** | **Status document** |
| 0.1 | 08-03-2021 | Lars van den Brandt | First draft | Waiting for feedback |

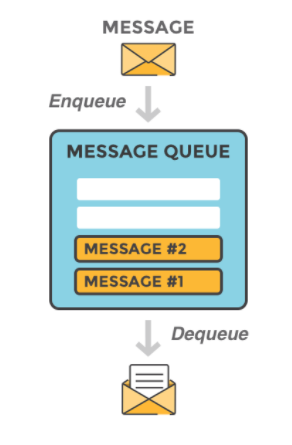
# Introductie

Dit document bevat een kort onderzoek naar RabbitMQ. Het onderzoek bevat verschillende vragen die zijn beantwoord met bronnen gevonden op het internet.

Het doel van dit onderzoek is het verkrijgen van kennis over RabbitMQ, om dit vervolgens te verwerken in mijn eigen project; ‘PolarGram’.

# Rabbit mq

## Wat is het?

RabbitMQ is een message-quing software. Het staat ook wel bekend als een message broker of een queue manager.   
RabbitMQ is in het kort: het is software waarin de wachtrijen worden gedefinieerd, waarmee applicaties verbinding maken om berichten te versturen.

Een bericht kan verschillende soorten informatie bevatten. Zo kan het informatie over een proces of een taak bevatten, maar het kan ook een simpel tekst bericht zijn. De software (RabbitMQ, een queue-managing software) slaat het bericht tijdelijk op. Hij stuurt het bericht weer verder door op het moment dat de ontvangende applicatie met de queue-managing software verbind en het bericht van de ‘wachtrij’ afhaalt.   
De berichten die worden opgeslagen op de queue-managing software worden dus opgeslagen tot een ontvangende partij klaar is om hem op te vangen.

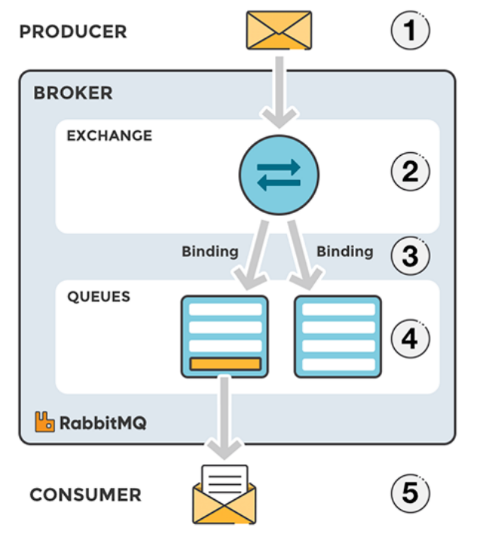
## waarom gebruiken?

Message queueing stelt webservers in staat om snel op verzoeken te reageren, in plaats van gedwongen te worden om resource-intensieve producers ter plekken uit te voeren, die bijvoorbeeld de responsetijd van de applicatie kan vertragen.   
Message queueing kan ook gebruikt worden om verschillende berichten tussen meerdere gebruikers te verdelen, zo kan de ‘work-load/balance’ worden verdeeld.

## hoe werkt rabbitmq?

De berichten/messages worden niet direct in de wachtrij gezet. Er wordt gebruik gemaakt van een ‘exchange’. De exchange ontvangt de berichten en plaatst de berichten in verschillende wachtrijen, die zijn verbonden aan bindings en routing keys, zo kan de juiste wachtrij later worden teruggevonden.

**Workflow:**

**1.**De ‘producer’ stuurt een bericht naar de exchange. Hierbij moet een type worden neergezet.

**2.** Wanneer de exchange het bericht ontvangt, koppelt hij er een routing key aan. Welke key dit is, hangt af van het type dat de producer heeft meegegeven.

**3.** Bindings moeten worden aangemaakt om het bericht in de juiste wachtrij te zetten.

**4.** Nu de berichten in de wachtrij staan, moet er worden gewacht op een bericht van de ontvanger. Tot die tijd blijven ze in de wachtrij staan.

**5.** De ontvanger ontvangt het bericht.

## wat zijn alternatieven voor Rabbitmq?

### Kafka

|  |  |
| --- | --- |
| **Voordelen** | **Nadelen** |
| High-throughput | Non-java clients are second-class citizens |
| Distributed | Needs Zookeeper |
| Scalable | Operational difficulties |
| High performance |  |

### ActiveMQ

|  |  |
| --- | --- |
| **Voordelen** | **Nadelen** |
| Easy to use | Support |
| Open source | Low resilience to exceptions and interruptions |
| Efficient | Difficult to scale |
| Scalable |  |

### Zeromq

|  |  |
| --- | --- |
| **Voordelen** | **Nadelen** |
| Fast | No message durability |
| Lightweight | Not a very reliable system |
| Transport agnostic |  |
| Open source |  |

# Conclusie

Als conclusie heb ik besloten om RabbitMQ te verwerken in mijn persoonlijke project. Het is een message broker die veel wordt gebruikt waardoor er veel informatie/tutorials over te vinden zijn. Verder gaan wij dit ook in ons groepsproject implementeren waardoor ik eventuele problemen samen met mijn groepsgenoten zou kunnen fixen.

# Bronnen

Bron 1: <https://www.cloudamqp.com/blog/part1-rabbitmq-for-beginners-what-is-rabbitmq.html>

Bron 2: <https://stackshare.io/rabbitmq/alternatives>

Bon 3: <https://www.rabbitmq.com>

Bron 4: <https://en.wikipedia.org/wiki/RabbitMQ>