What’s RabbitMQ?

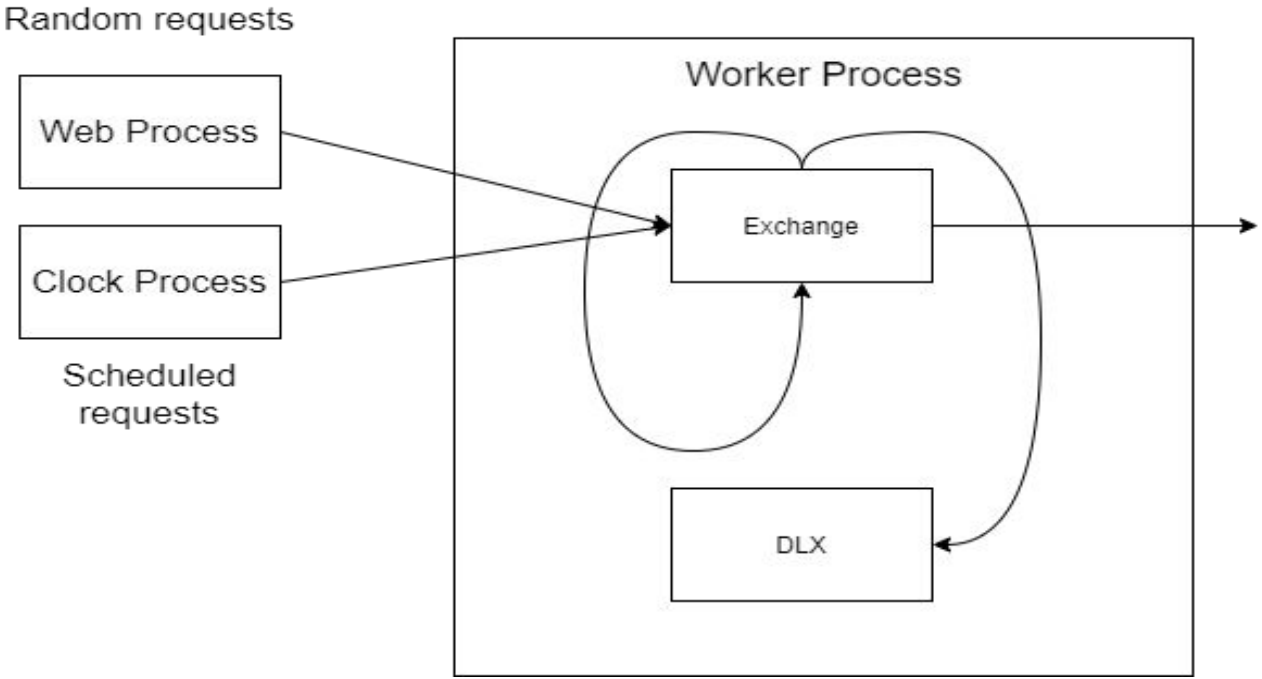
RabbitMQ is an open source message broker that supports many protocols.

What does RabbitMQ do then?

- It will receive messages and put them in a queue
- A queue will wait for a consumer to connect before giving it its stored messages, one-by-one
- The queue requires the consumer to confirm that the message has been read and can be discarded
- If a consumer acknowledges having read a message, but rejects it, the queue will dead-letter it, leaving the message available for future processing

These points are important because it means you can ensure the safety of your data and that your data will undergo some process of your choosing.

How does RabbitMQ work?



Record/Service

If the message was processed successfully, it probably went to an outside service for record keeping, otherwise it went in the DLX, awaiting further processing.

A message-driven mechanism opens up possibilities

With a message broker, we can open up scheduled jobs to on-demand, meaning an endpoint can be exposed to trigger jobs that would normally only happen at a specified schedule.

It can also let us integrate with partners (internal and external) more easily; letting us serve our client as they're used to until we're sure integration with the partner is rock solid.

It can move process-intensive tasks from the in-demand API to a worker, or a cluster of such processes.