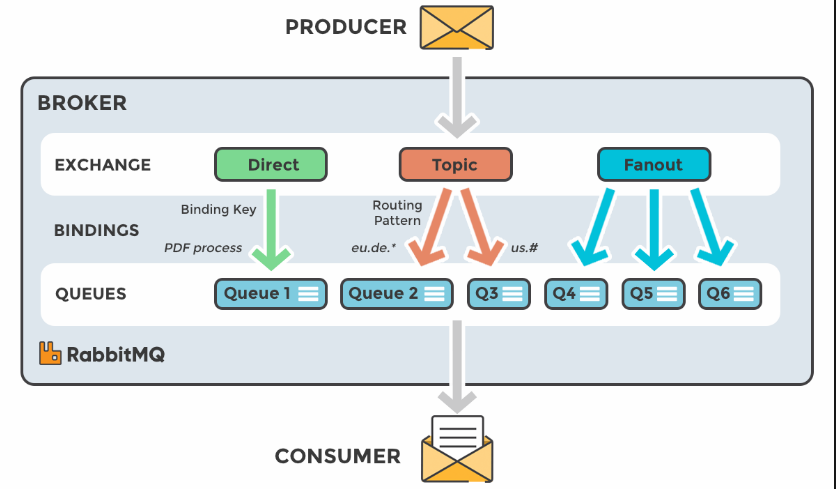
**RabbitMQ**

RabbitMq is message queue software also called message broker or queue manager ,a way of exchanging a data between processes, application components and server.

RabbitMq has written in Erlang and has driver/clients available for most major language.

Or

Simply it said ,It is software where queues can be defined,application may connect with queue transfer a message onto it.



We can consider RabbitMQ as post-box where we drop letter to post-box and right person deliver it to recipient.In same way RabbitMQ do it where RabbitMQ itself act as post-oofice,post-box and post master.means that it will accept message store the message and deliver the message.

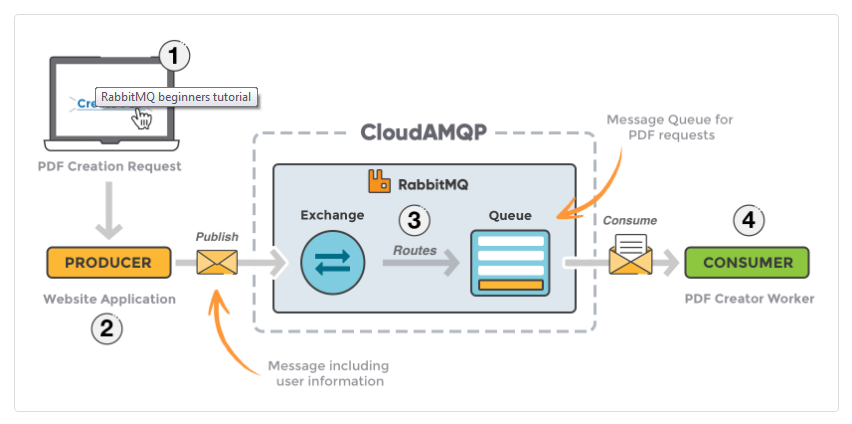
A message can include any kind of information. It could, for example, have information about a process/task that should start on another application (that could be on another server), or it could be just a simple text message. The queue-manager software stores the messages until a receiving application connects and takes a message off the queue. The receiving application then processes the message in an appropriate manner.

When and why should you use RabbitMQ?

When we want a server to give quick response on the request .and Message queue of RabbitMQ is also preferable when we want to sent message to multiple recipient or for balancing load between workers.

Client or publisher and consumer can be on same server or different server to communicate each other through message queue.

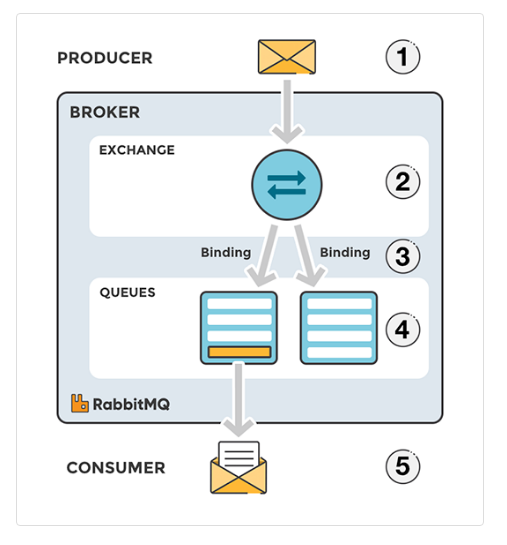
In RabbitMQ ,An Application running on one server can communicate with another application on same/different server through RabbitMQ.



1. The user sends a PDF creation request to the web application.
2. The web application (the producer) sends a message to RabbitMQ, including data from the request, like name and email.
3. An exchange accepts the messages from a producer application and routes them to correct message queues for PDF creation.
4. The PDF processing worker (the consumer) receives the task and starts the processing of the PDF.

#### EXCHANGES

Messages are not published directly to a queue, instead, the producer sends messages to an exchange. An exchange is responsible for the routing of the messages to the different queues. An exchange accepts messages from the producer application and routes them to message queues with the help of bindings and routing keys. A binding is a link between a queue and an exchange.



\_\_\_\_\_\_\_\_\_

Broker(message queue)

Client(producer/sender)

For installation of RabbitMq

<https://www.rabbitmq.com/install-windows.html>

<http://www.erlang.org/downloads>

sending a message is nothing but producer and receiving a message is consumer.

Queue is the name of post-box which lives inside RabbitMq. Although message flow through RabbitMQ and your application.

Many producer can send a messages to one queue and many consumer can read message from one queue.