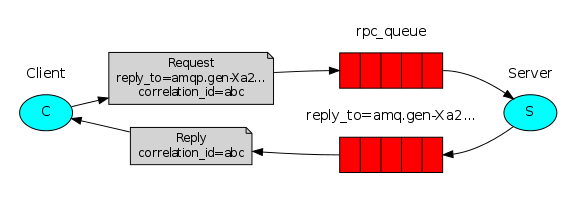
# RPC System (Node js web api + C# server)

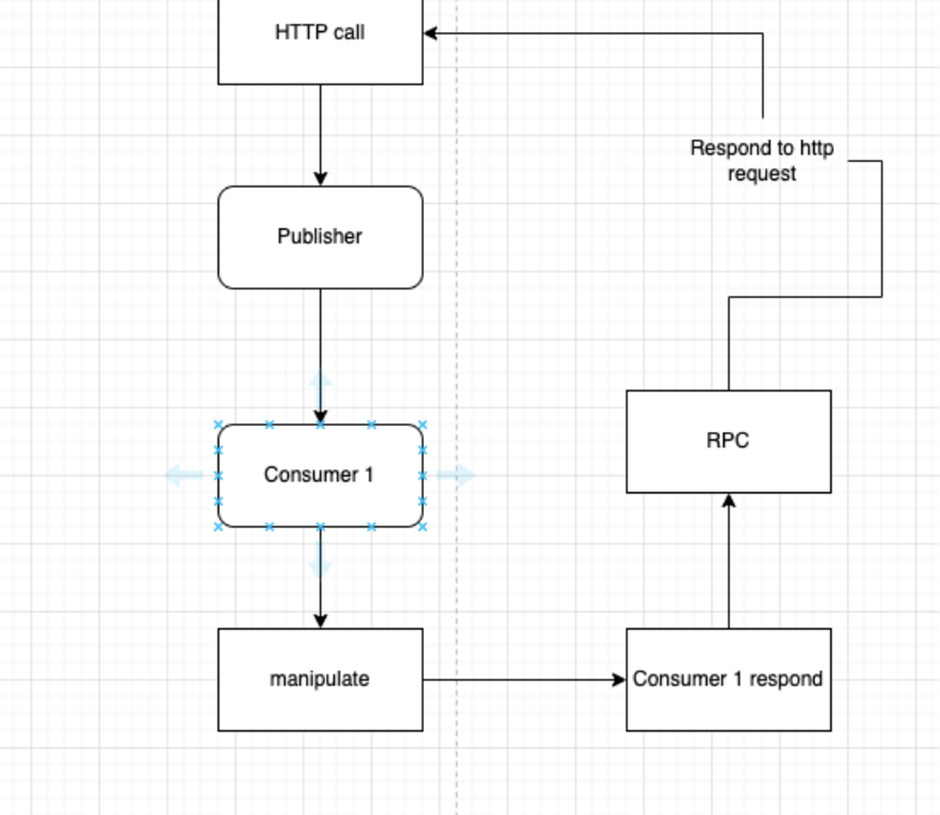
Create client server-based architecture in which you have to use RabbitMQ to build an RPC system: a client and a scalable RPC server.

## Task – Client needs to request a data from server, Server needs to fetch data from database according to request type.



## **Basic Details –**

* Client should be node js web api which can be easily called from postman.
* Server should be in c# .
* Client will communicate to server through an RabbitMQ queue.
* Database can be anything.(SQL,Notepad,Xml).



# **Major Problems –**

* Server may need some time to process and send response back to client, but request object always behaves in synchronous way, which can break RPC system, so you have to find a way

to hold request object till server responds.

* Generally, every time client receives request it creates temporary RabbitMQ channel and destroy it when cycle completes, which is less efficient, so you have to find a way in which do not need to create channel every time.
* Every request is unique so is response, so there should be no mismatch,

Ex- There are 10 requests in queue waiting for response, server need to identify which response needs to be sent to which request, there should be 0 errors expected.