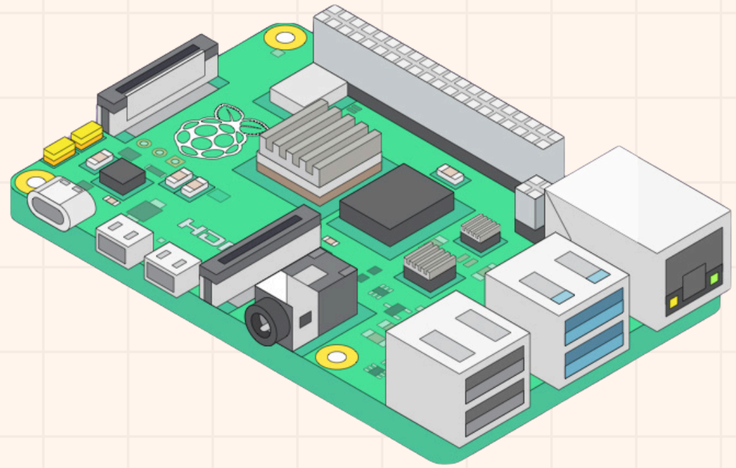
SIMPLIFYING **IOT**
COMMUNICATION

MQTT
RASPBERRY PI

MQTT



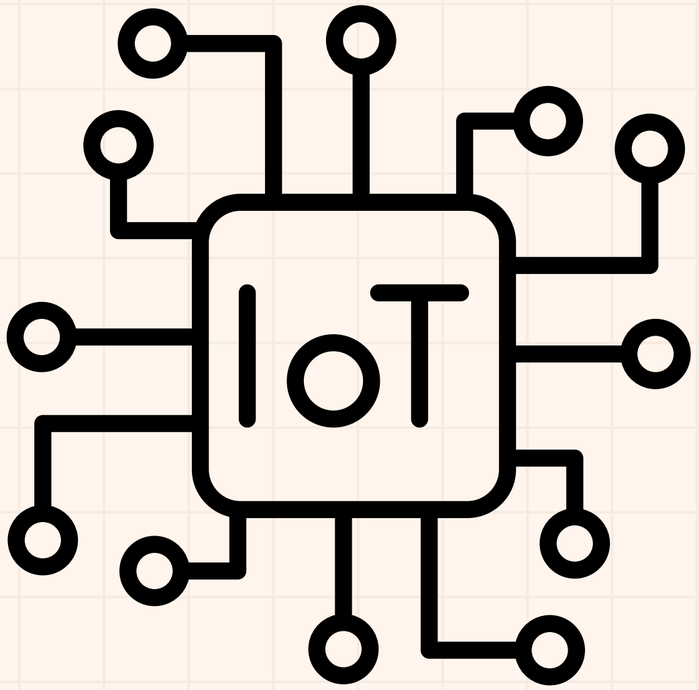
MQTT (Message Queuing Telemetry Transport) — the lightweight communication protocol that's perfect for IoT projects.

ASHISH JOY

IOT - INTERNET OF THINGS

A network of physical devices ("things") connected to the Internet, collecting and exchanging data without human intervention.

IoT devices—also known as “smart objects”—can range from simple “smart home” devices like smart thermostats, to wearables like smartwatches and RFID-enabled clothing, to complex industrial machinery and transportation systems. Technologists are even envisioning entire “smart cities” predicated on IoT technologies.



ASHISH JOY

COMMUNICATION
PROTOCOLS

MQTT

Communication protocols are the set of rules that allow electronic devices to exchange data effectively. In the world of IoT (Internet of Things), these protocols ensure that sensors, microcontrollers, and servers can talk to each other reliably, whether they are nearby or spread across the globe.

Some popular communication protocols used in IoT include **MQTT** (Message Queuing Telemetry Transport), **HTTP**, **CoAP**, **WebSockets**, **Bluetooth Low Energy** (BLE), **Zigbee**, **LoRaWAN**, **Wi-Fi**, and **Cellular** (4G/5G, NB-IoT).

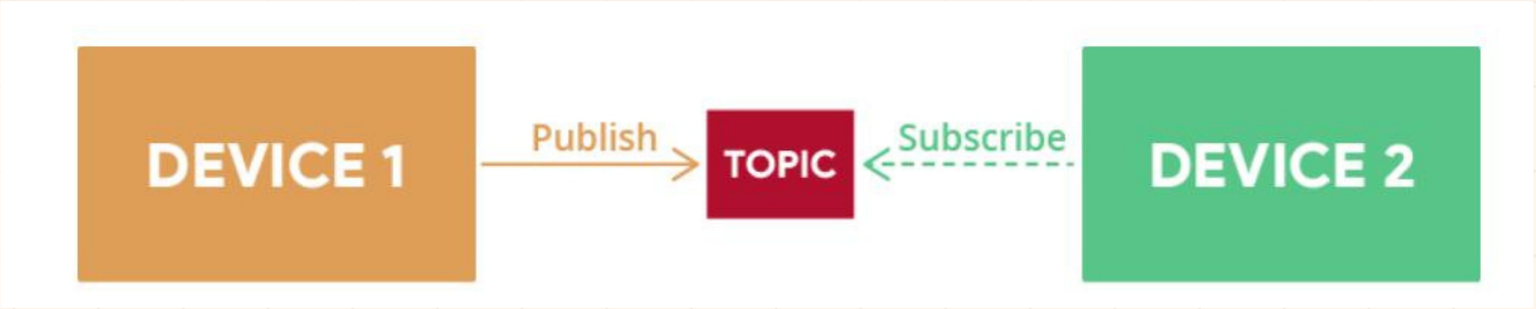
ASHISH JOY

MQTT

MESSAGE QUEUING TELEMETRY TRANSPORT

MQTT stands for Message Queuing Telemetry Transport. MQTT is a simple messaging protocol, designed for constrained devices with low bandwidth. So, it's the perfect solution to exchange data between multiple IoT devices.

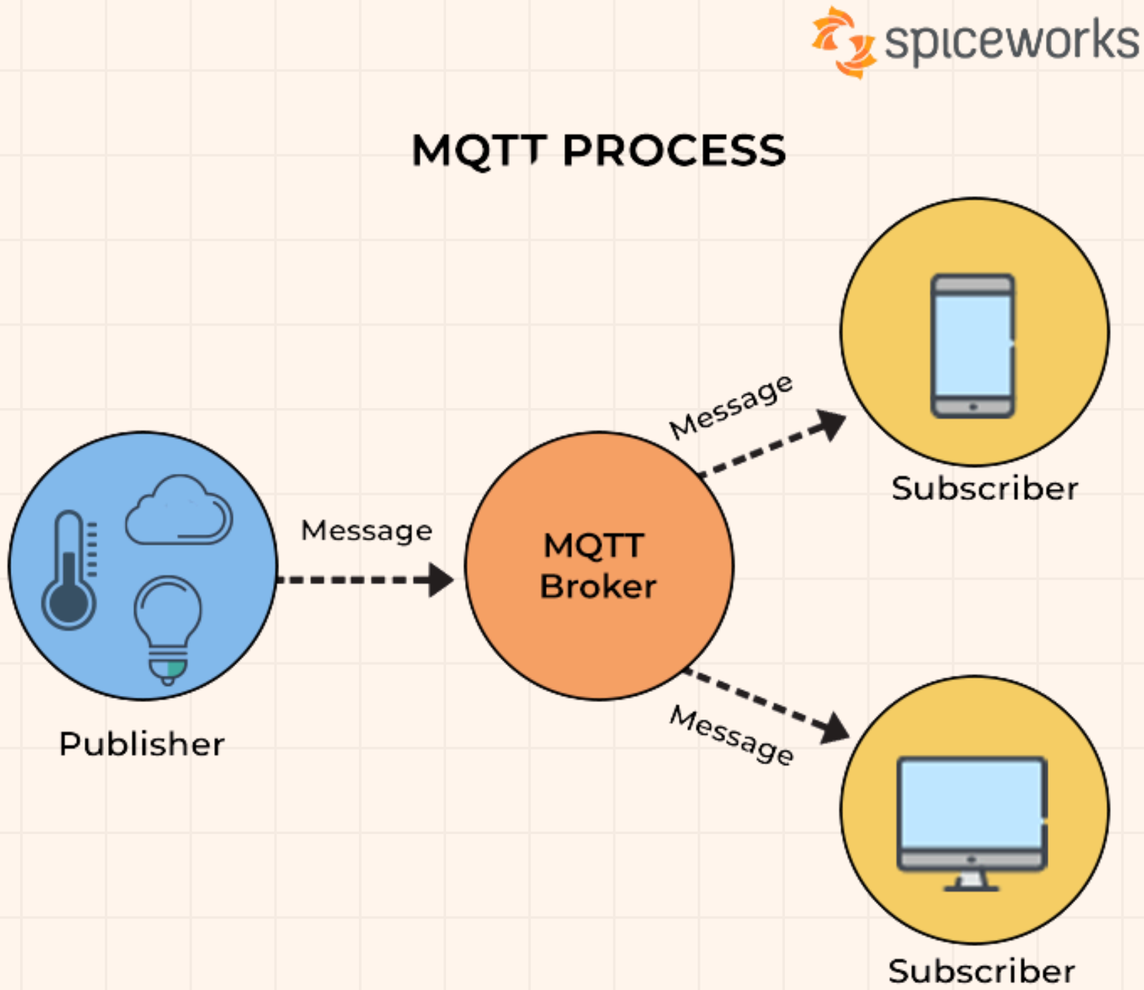
MQTT communication works as a publish and subscribe system. Devices publish messages on a specific topic. All devices that are subscribed to that topic receive the message.



ASHISH JOY

MQTT
BASIC CONCEPTS

- PUBLISH/SUBSCRIBE
- BROKER
- TOPIC
- PAYLOAD



ASHISH JOY

SUBSCRIBE AND PUBLISH

PUBLISHER → BROKER → SUBSCRIBER

DEVICES DON'T TALK TO EACH OTHER DIRECTLY.
INSTEAD, THEY USE A **BROKER** (MEDIATOR) TO PASS MESSAGES.

PUBLISHER SPEAKS ON THE TOPIC
SUBSCRIBER IS LISTENING ON THE TOPIC

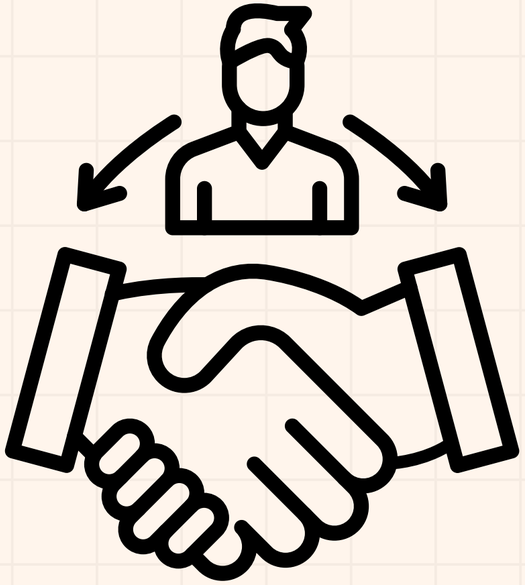
A SINGLE DEVICE(CLIENT) CAN BOTH ACT AS A **PUBLISHER** AND **SUBSCRIBER**

ASHISH JOY

BROKER

A Broker is the central server that manages all the messages between Publishers and Subscribers in the MQTT protocol.

Eg. Mosquitto, HiveMQ

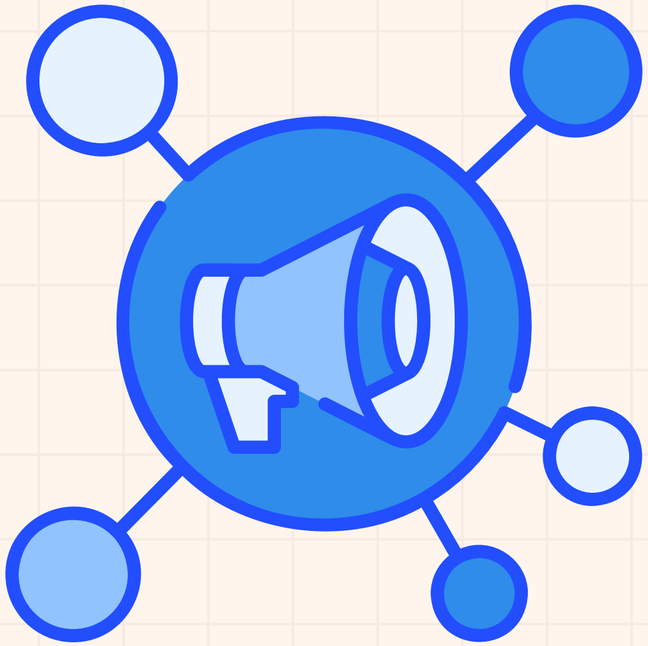


TOPICS

A Topic is like a **label** or **channel** name used by MQTT to organize and route messages between devices.

For. eg. A telegram channel. If you are subscribed to a telegram channel you can see the messages in the channel. And you can publish messages to the group.

It helps the Broker know where to deliver each message.



A Topic can be something like **home/AC** or **tinkerspace/hackerspace/lights**

PAYLOAD/MESSAGES

A Payload is the actual data or message content sent by a Publisher and received by a Subscriber in MQTT communication.

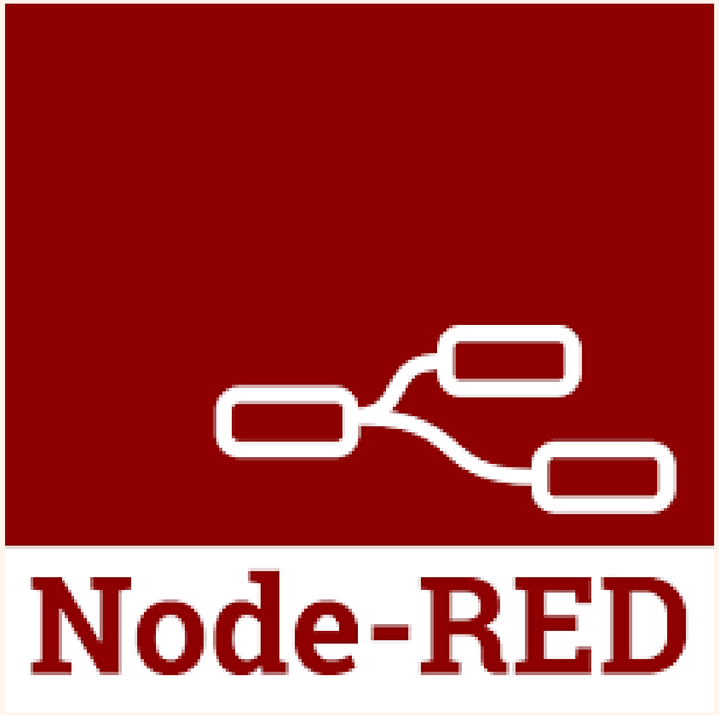
Think of it as the "letter" inside an envelope, where the Topic is the address, and the Payload is the message!

NODE RED

NODE-RED IS A VISUAL TOOL FOR WIRING THE INTERNET OF THING (EMBEDDED SYSTEM AND COMMUNICATIONS)

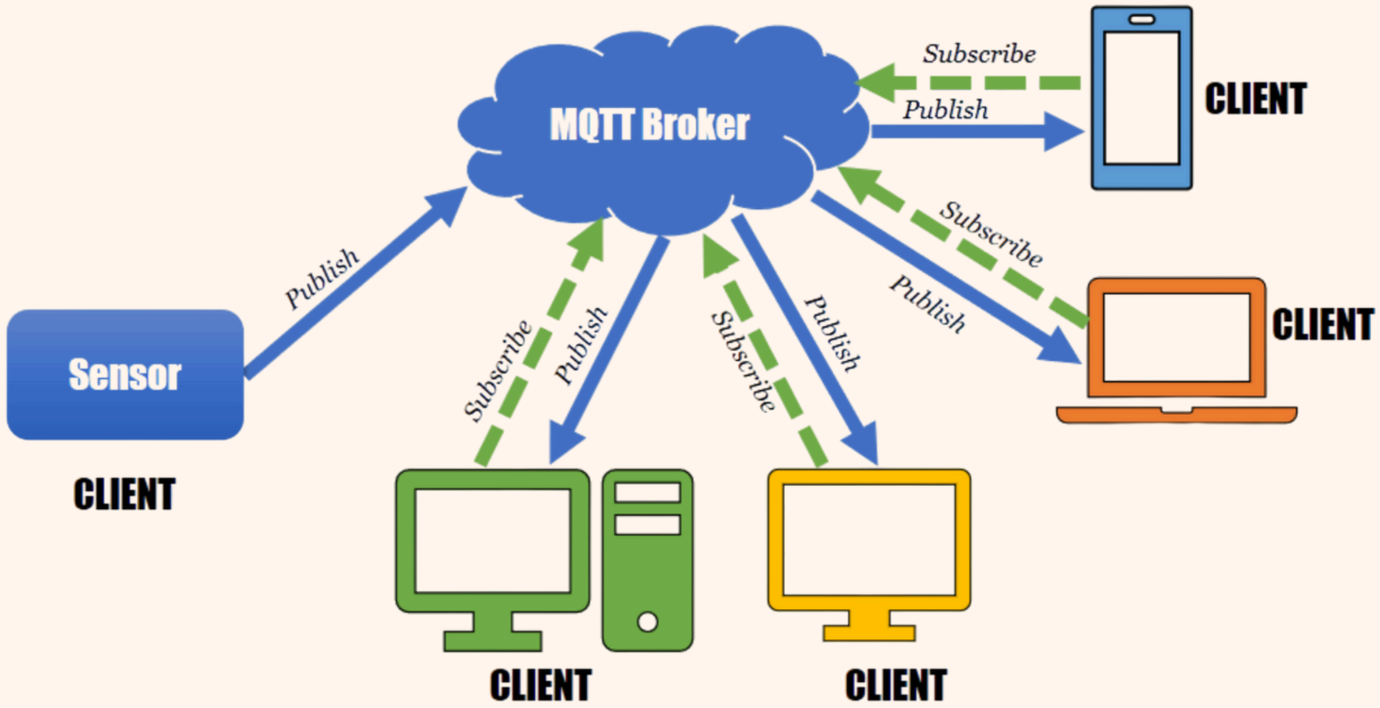
BASED ON NODE JS

NODE-RED IS AVAILABLE AS OPEN SOURCE AND HAS BEEN IMPLEMENTED BY THE IBM .



ASHISH JOY

NODE RED
ON RASPBERRY PI



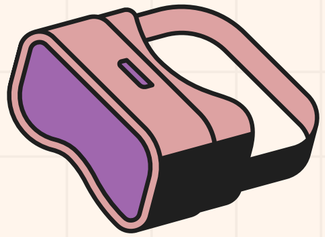
Mosquitto Broker

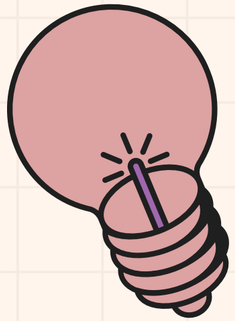
SUDO APT UPDATE
SUDO APT INSTALL MOSQUITTO MOSQUITTO-CLIENTS
SUDO SYSTEMCTL ENABLE MOSQUITTO
SUDO SYSTEMCTL START MOSQUITTO

MOSQUITTO_SUB -H LOCALHOST -T TEST/TOPIC

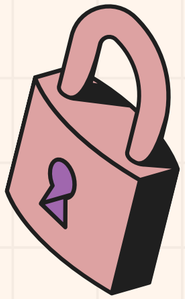
MOSQUITTO_PUB -H LOCALHOST - T TEST/TOPIC "HELLO"

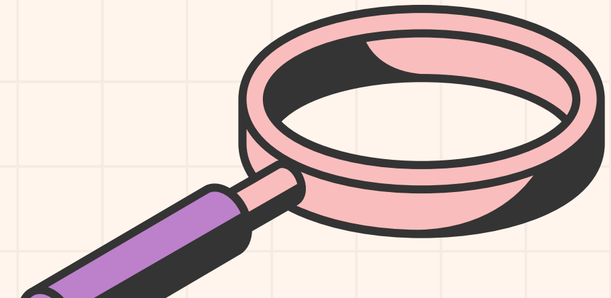
ASHISH JOY





THANK YOU





ASHISH JOY