

# Data Analysis in Io(B)T

**Mystica**  
Software Developer

Find me online  
[mystica.me](http://mystica.me)



# Agenda

- Introduction
- Usecases
- Architecting IoT Apps
- MQTT Protocol
- Demo
- Applications
- More Applications
- Security
- Analysis and Conclusion

# Why do we even need Data Analytics in IoT?

Important values of IoT?

- collect data from tons of sensors or other microcontrollers.
- new service can be integrated via those devices (Alexa integration)
- business insights we can gather from those data

- Real-world data is messy and will be large too.
- The more IoT devices added to the networks, the more the data will be. It can be overwhelming for a system to handle such an amount of data in a small period.
- so we go for "Big Data Analytics" when our product/system scales.

# Use case #1

## Spot by Boston Dynamics

Robot + AI

# Customize

- Add Models/Sensors
- Collect data from SPOT API (Store for later / real time)
- Analyze



# Case Studies

- Thermal anomaly detection
- NASA BRAILLE
- Nuclear Power Plants
- Measures temperature from equipment
- High risk radiation areas



"sense the world and interact  
with it!"

Contains Actuators and Sensors

For who it will be useful??

- High risk operations
- Going somewhere no human can go (deep cave)

# Defense Applications

## Case Studies

### Lockheed Martin

- Search and Rescue
- Humanitarian Crisis
- Medical Supply



# Predictive Modelling

- Maintenance and repairing of airlines

## Data Might Includes

- Fuel Level
- Weather
- Equipment Maintenance Record

# Architecting IoBt Applications

No "Silver Bullet"

# MQTT Protocol

- MQ Telemetry Transportation
- Lightweight Messaging Protocol
- based on pub-sub model

MQTT Broker  
MQTT Client

# Pub-Sub

- Publisher - Subscriber(Producer-Consumer) Model
- Used in lots of modern applications in software development and cloud
- Asynchronous (non-blocking)

# Why do we need pub-sub

Application  
(B) that  
needs the  
data from A

Application  
(C) that  
needs the  
data from A

Application  
(D) that  
needs the  
data from A

Application(A)  
that Produces  
the  
Data

Application  
(B) that  
needs the  
data from A

Application  
(C) that  
needs the  
data from A

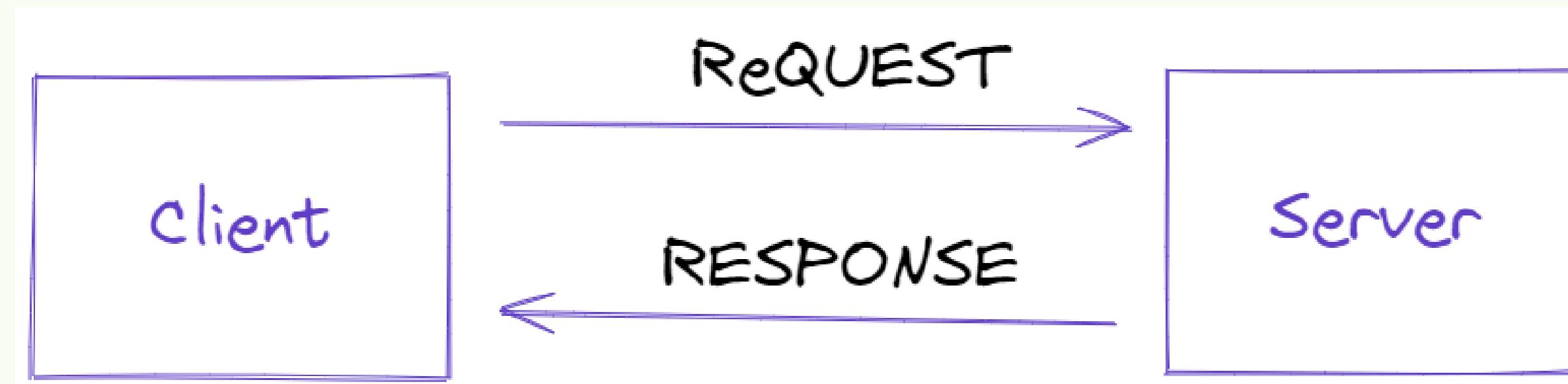
Application  
(D) that  
needs the  
data from A

Application(A)  
that Produces  
the  
Data

Change in data present in  
Application (A) have to be broadcasted  
to B,C,D

# What's Protocol we can use?!

HTTP??

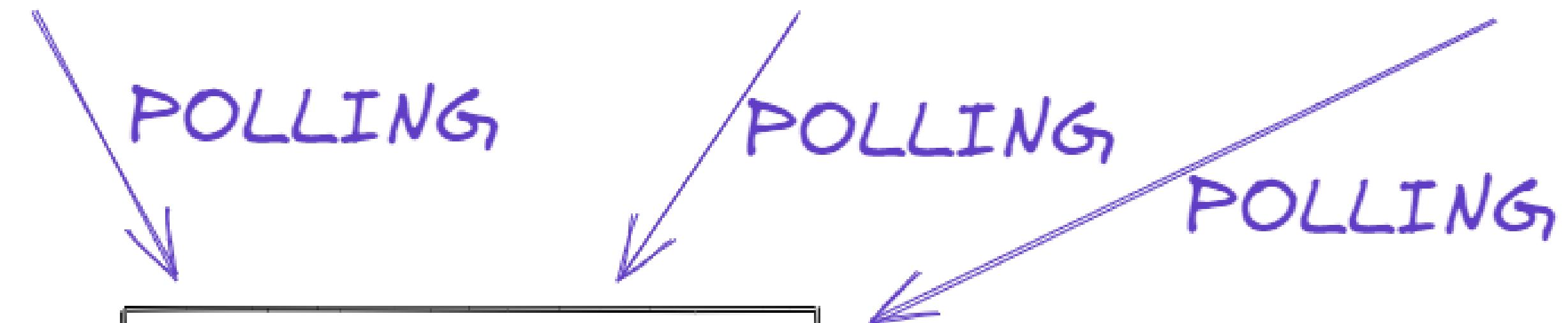


- HTTP is not designed for bi-directional communications.
- Client can request Data from the Server and get Response Back, that's the end of the connection!
- What if i want the server to send the data to the client after some time (sensors measures every 1hr)

Application  
(B) that  
needs the  
data from A

Application  
(C) that  
needs the  
data from A

Application  
(D) that  
needs the  
data from A



Application(A)  
that Produces  
the  
Data

Change in data present in  
Application (A) have to be broadcasted  
to B,C,D

# HTTP won't be a good choice here!

- We are hacking our way to make things works for us..

As the system grows, it's not easier to maintain

We need a bidirectional  
way of communicating

Client A: \*requests the server\*

Server : \*receives the request, registers the client by unique ID\*

Client A: Whenever you have new data, update me!

**\*\*For every new Data\*\***

Server : \*sends the data to the client (how? using client id) \*

# That's Pub-Sub

Whenever the server publishes the data, client gets!

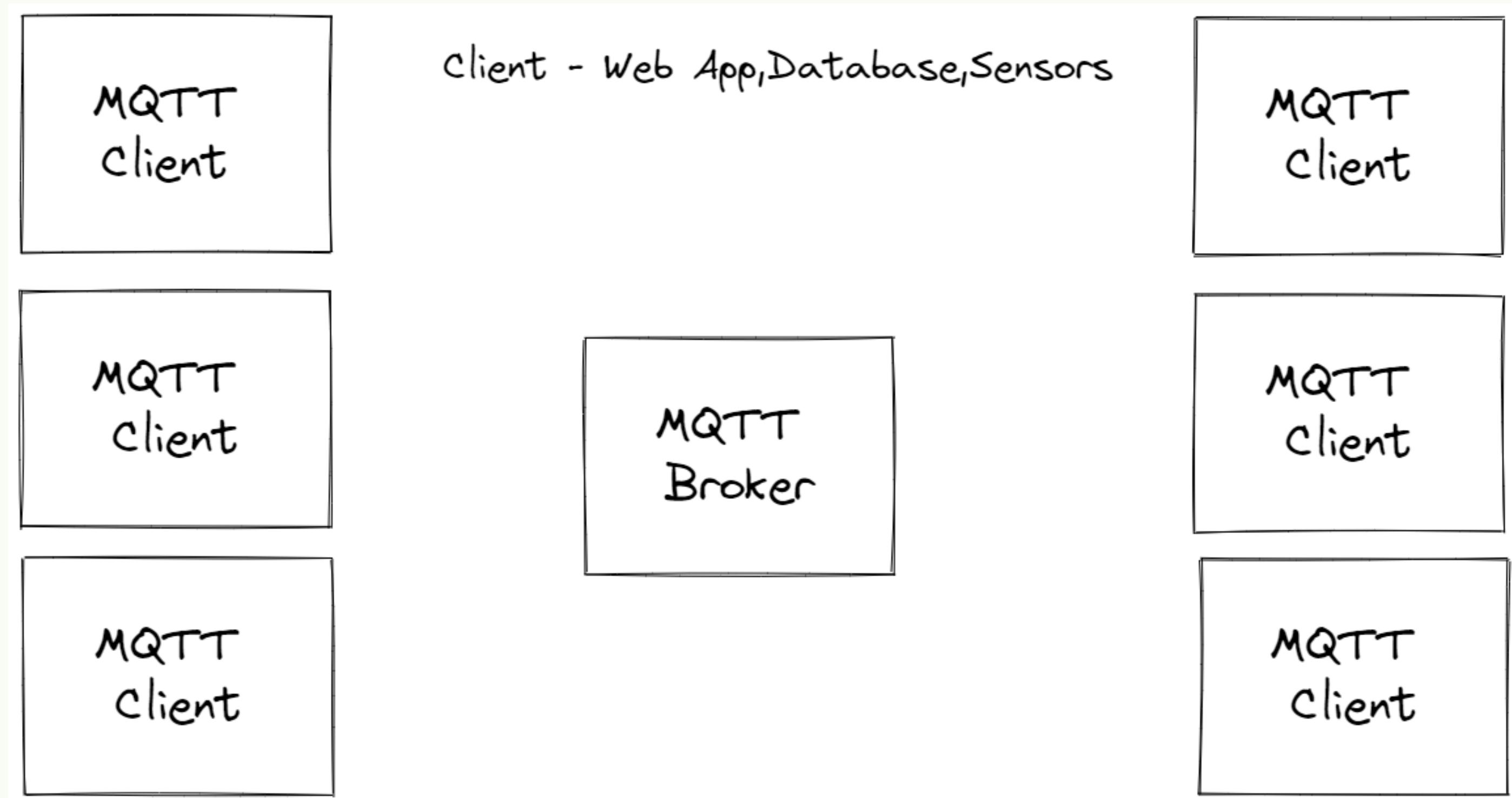
Server - Publisher

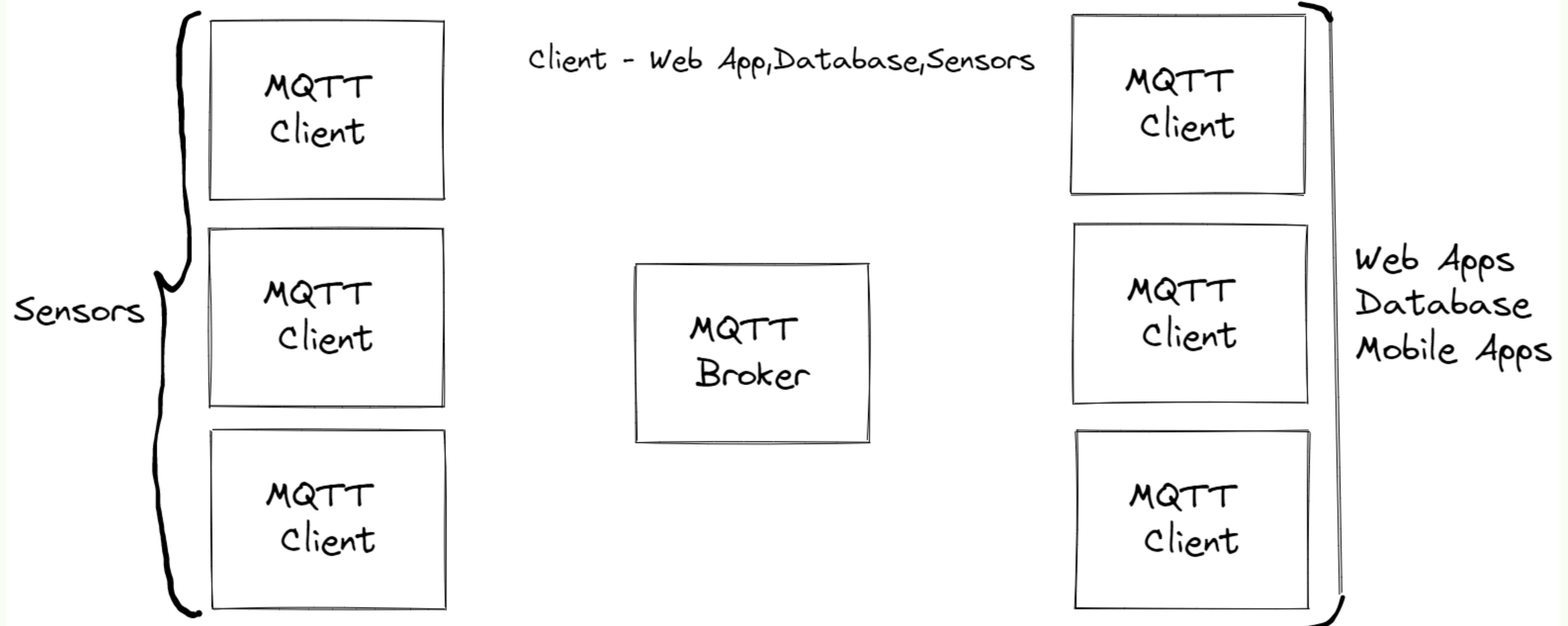
Client - Subscriber

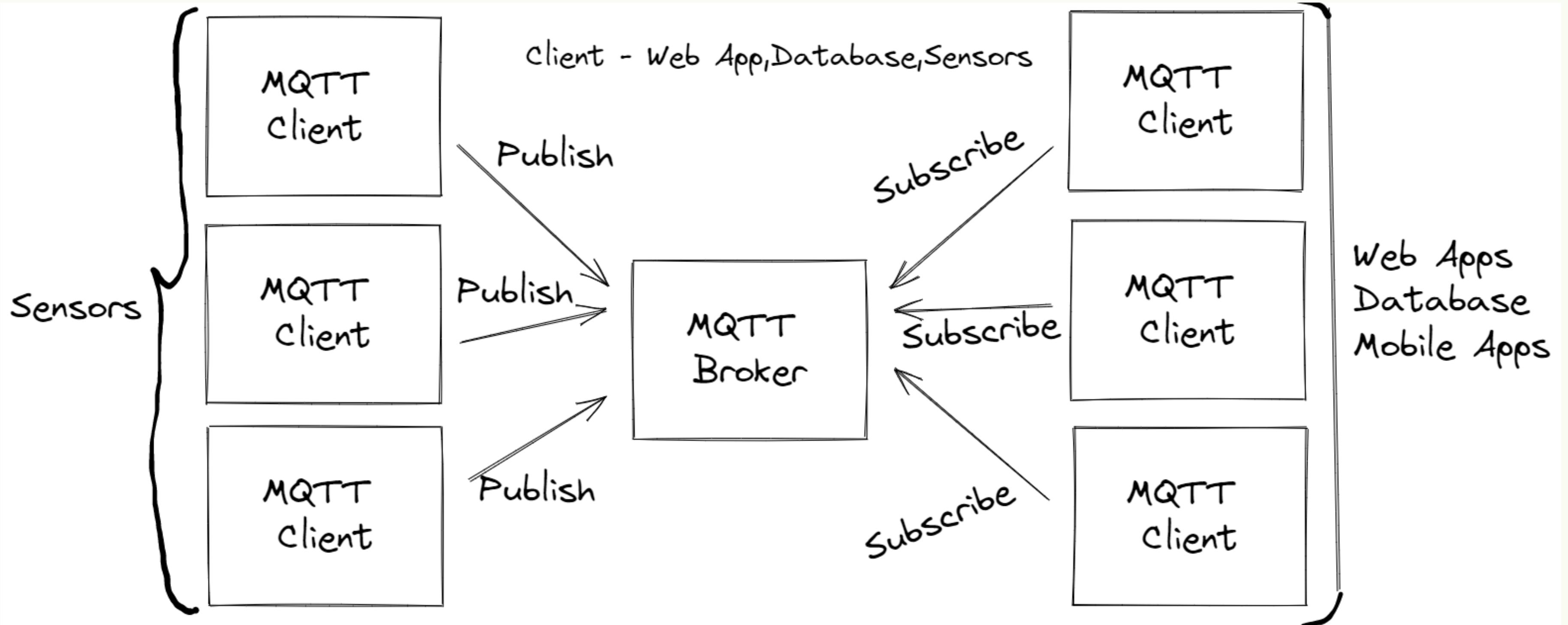
# Some examples of where this pattern is used!

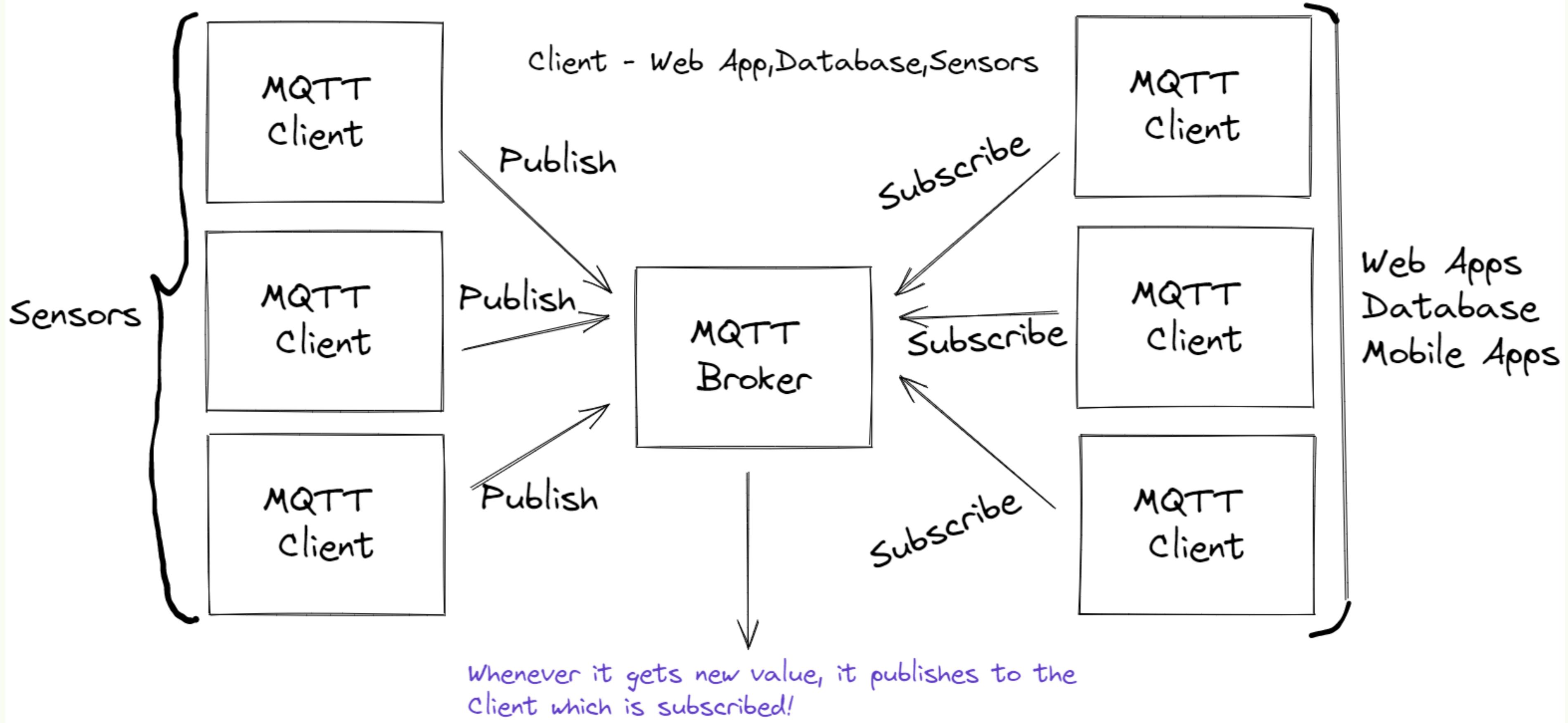
- In Frontend Apps, RxJS (Reactive)
- Microservices
- Cloud Data
- IoT messaging

# Back to MQTT,









Broker acts a central hub that manages the clients

# Now, we seen sensors as MQTT Client

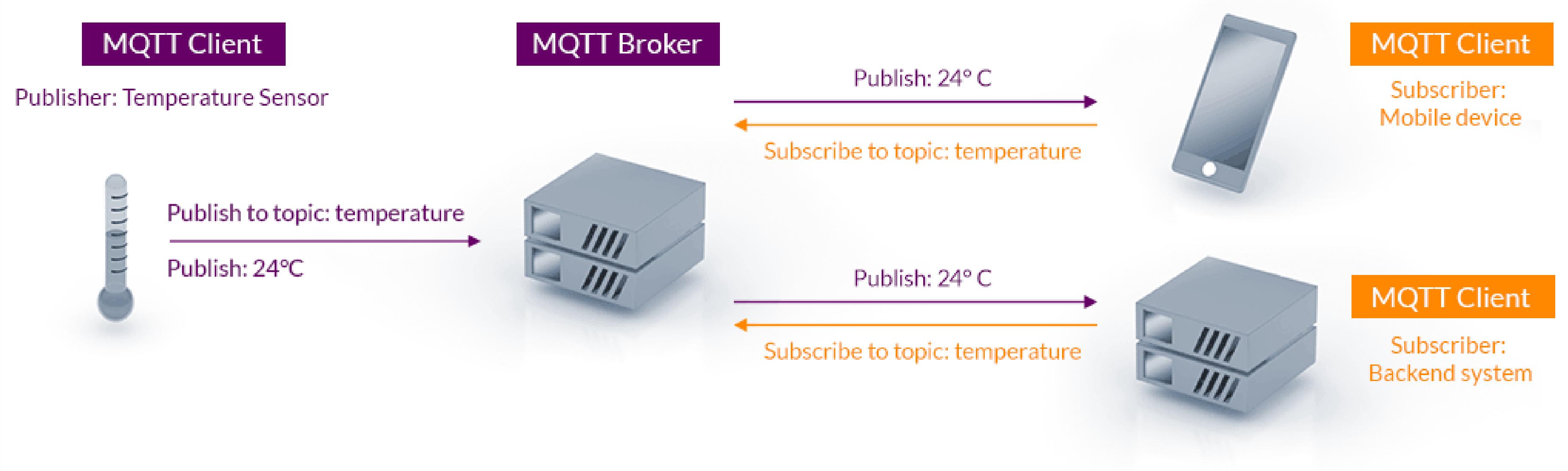
- What if One Client sends Temperature data, Other sends Sound level data, yet another sends image data.
- They all will be publishing values to the broker node! how can we differentiate this data!...

# Now, we seen sensors as MQTT Client

- What if One Client sends Temperature data, Other sends Sound level data, yet another sends image data.
- They all will be publishing values to the broker node! how can we differentiate this data!...

"Topic"

# From official docs, (mqqt.org)



Differentiate based on "Topic"

Why do we need broker?? we can  
send directly to the DB or  
server,right?

Why do we need broker?? we can  
send directly to the DB or  
server,right?

lot or real time lot, we will be use "Edge" a LOT!!

Moving nearer to the USER/Device!

If we use cloud, latency and bandwidth!

# MQTT Features

Out of the box, solution for lower network data!

It uses TCP/IP protocol under the hood!

ACK-SYNC BASED

CLIENT

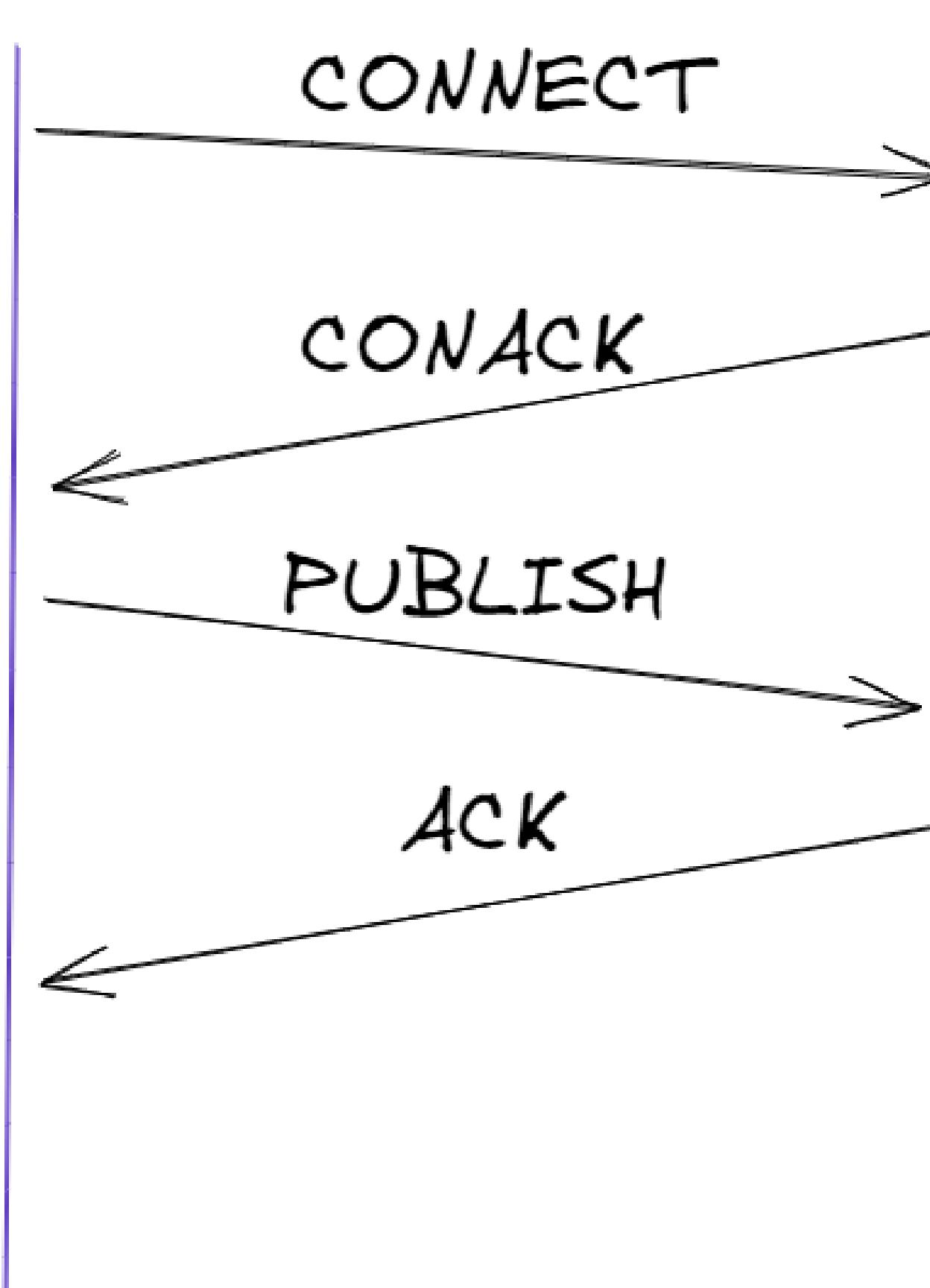
Broker

CONNECT

CONACK

PUBLISH

ACK



CLIENT

Broker

CONNECT

CONACK

PUBLISH

ACK

Callback

# Callback!

We can write a logic inside a callback function  
what to do after sending the data!

Example: data has been sent, sleep for 1hr and then again measure the readings, sent it to broker

What if our data got lost while sending it to the broker or due to some problem, broker cannot send the acknowledgement back to the client!

# QOS (Quality of Service)

- 0 - at most once (Dont send ack back to client)
- 1- at least once (Guarantees messages sent to broker)
- 2- exactly once (Duplicate Flag set if retransmitting again, so that broker has one set of data instead of all the duplicates)

# When to use Which QOS

- 0 - incase of stable connection!
- 1 - Commonly used, and guarantees delivery
- 3- Case where duplicate affects the user

# Anyone can send data to MQTT Broker?

## MQTT Authentication

- Username, Password
- TLS

# How to implement it?

**Server** - Software available for broker!

**Client** -

Wrapper for Clients in all modern languages

# Sample System Design for Enterprise level IoT App

## Problem statement:

Create an enterprise level application that reads the data from sensors all over the world, and store it on cloud for analysis

# What we will be using:

- MQTT Broker
- Sensors
- Database
- Server
- Data analytics

<https://excalidraw.com/>

# Kafka MQTT

Contains a plugin that receives MQTT data

MQTT is lightweight protocol that doesn't support data integration or can store data on its own!

Meanwhile, kafka can process data and publishes to various services and store data too

# Demo!

- Install Python!
- <https://pypi.org/project/paho-mqtt/>
- <https://www.emqx.io/>

# MQTT Broker

- EMQX Cloud Broker

# MQTT Client

- Python Wrapper

[https://github.com/mystica2000  
/MQTTDemo](https://github.com/mystica2000/MQTTDemo)



# Military Applications using Data Analytics

- Predictive Analysis (Aircraft maintenance)
- Target Detection using a Drone
- Mine Detection and Marking using robot
- Target Detection using radars
- Surveillance

# Military Applications using Data Analytics (contd..)

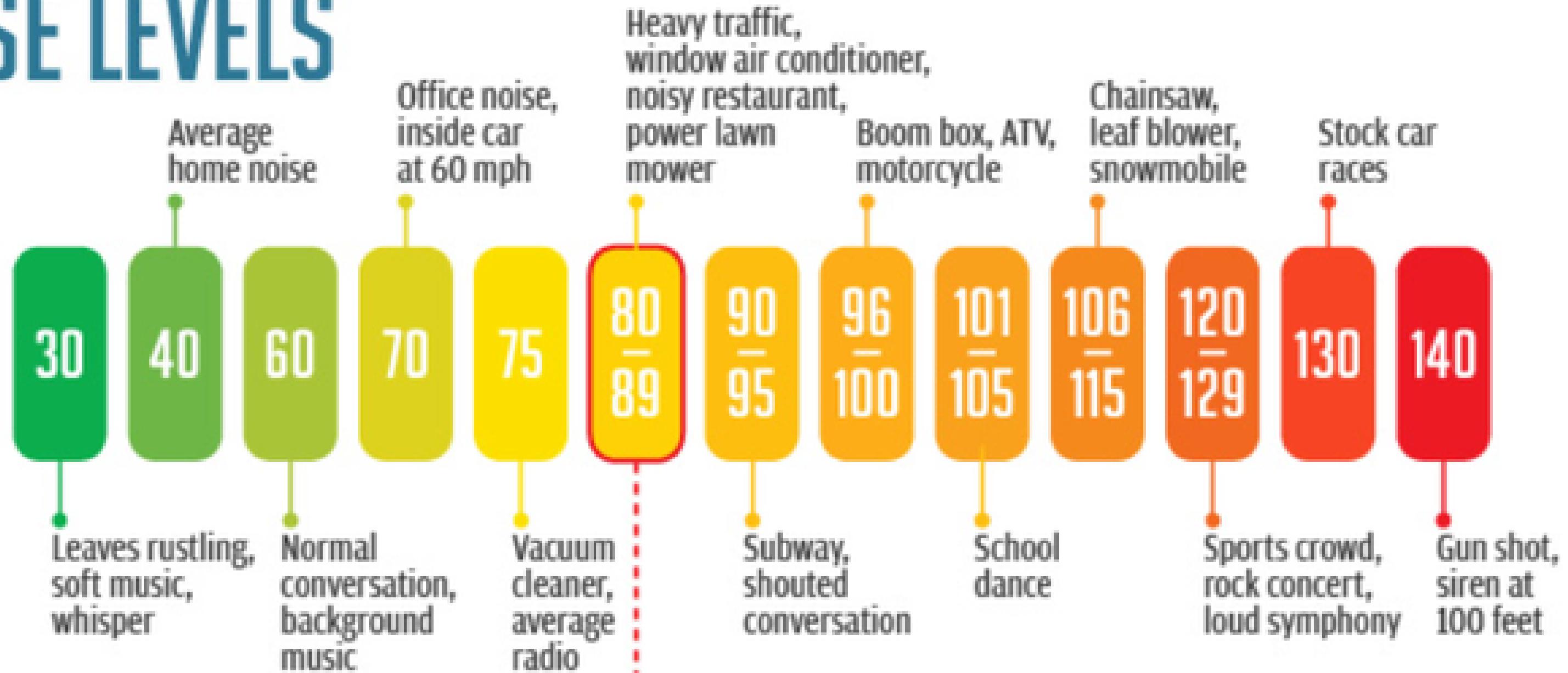
- Spying Devices
- Fake News Identifier
- Satellite image analysis

# Example: Spying!



1. Install Sound Detection Sensor
2. Monitor the decibel and perform analysis on the buildings.
3. Find unusual Detection of sounds and alert the officials about it!

# NOISE LEVELS



Source: [www.webmd.com](http://www.webmd.com)



**Sounds above 85 dB are harmful**

# Object Detection:

- Particle Filter Algorithm is used by Australian Military Forces when searching for missing airplane (MH 370)
- Unsuccessful one, because of the amount of area to search for was huge in the sea so they gave up that idea!

# Security in IOT

Unauthorized Access

Needs more authentication methods to  
prevent attackers

**The END!**